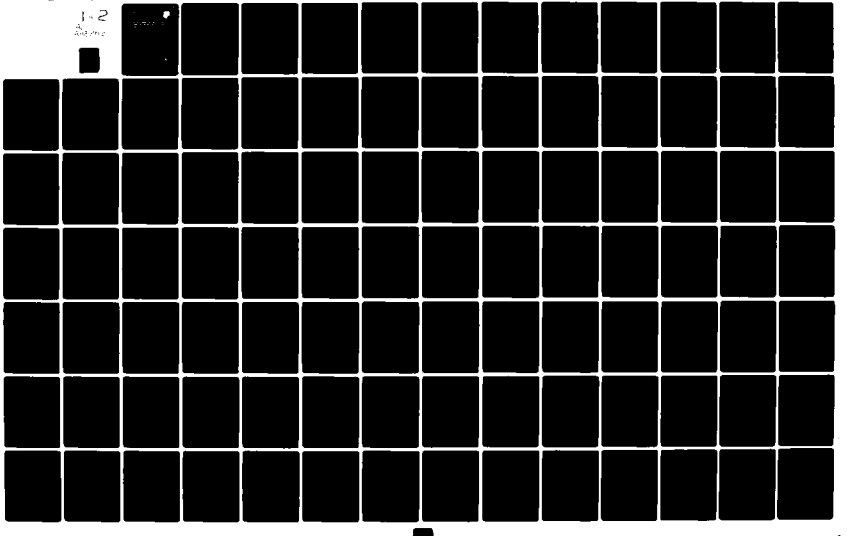
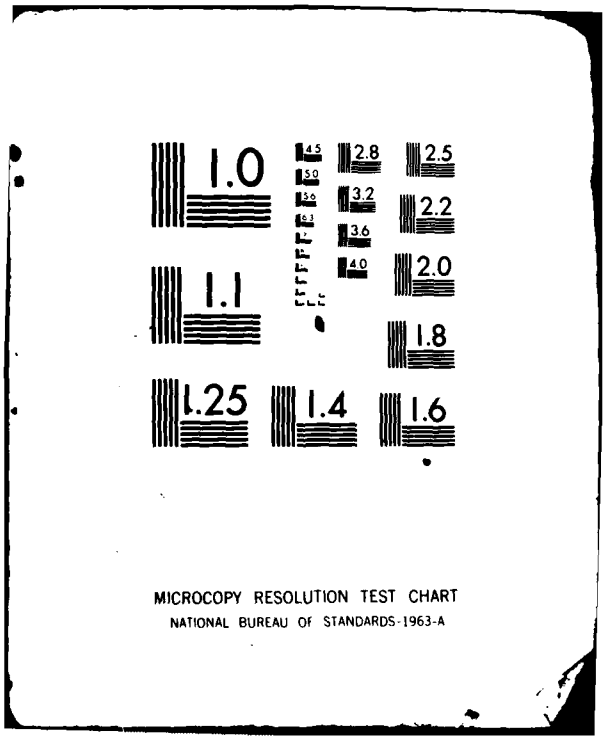
AD-A113 262 IIT RESEARCH INST ROME NY F/G 9/2
ESTABLISHMENT OF THE DATA AND ANALYSIS CENTER FOR SOFTWARE (DAC--ETC(U)
JAN 82 G CARON, D CORNWELL, J DOBMEIER F30602-78-C-0255
UNCLASSIFIED RADC-TR-81-385 NL




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD A113262

RADC-TR-81-385
Final Technical Report
January 1982

ESTABLISHMENT OF THE DATA AND ANALYSIS CENTER FOR SOFTWARE (DACS)

IIT Research Institute

Gary Caren
Douglas Cornwell
John Dahmeier
Lorraine Duvall
Shirley A. Glass-Soler
Christopher Turner

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, New York 13441

DTIC
ELECTE
APR 0 9 1982
E

82 04 09 057

This report has been reviewed by the RADC Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RADC-TR-81-385 has been reviewed and is approved for publication.

APPROVED: John Palaimo

JOHN PALAIMO
Project Engineer

APPROVED:

JOHN J. MARCINIAK, Col, USAF
Chief, Command and Control Division

FOR THE COMMANDER: John P. Huss

JOHN P. HUSS
Acting Chief, Plans Office

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (COER) Griffiss AFB NY 13441. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document requires that it be returned.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: RADC-TR-81-385	2. GOVT ACCESSION NO.: AD-A113262
	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): ESTABLISHMENT OF THE DATA AND ANALYSIS CENTER FOR SOFTWARE (DACS)	5. TYPE OF REPORT & PERIOD COVERED: Final Technical Report Aug 78 - Aug 81
	6. PERFORMING ORG. REPORT NUMBER: N/A
7. AUTHOR(s): Gary Caron, Douglas Cornwell, John Dobmeier, Lorraine Duvall, Shirley A. Gloss-Soler, Christopher Turner	8. CONTRACT OR GRANT NUMBER(s): F30602-78-C-0255
9. PERFORMING ORGANIZATION NAME AND ADDRESS: IIT Research Institute/Beeches Carriage Suite, Turin Road, Rome NY 13440	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 63728F, 25280102
11. CONTROLLING OFFICE NAME AND ADDRESS: Rome Air Development Center (COEE), Griffiss AFB NY 13441	12. REPORT DATE: January 1982
	13. NUMBER OF PAGES: 186
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office): Same	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE: N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Same

18. SUPPLEMENTARY NOTES

RADC Project Engineer: John Palaimo (COEE)

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Software Engineering
Information Analysis Center
Database
Scientific and Technical Information
Software Technology

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The Data & Analysis Center for Software (DACS) has been established to serve as a central source for information and data on software technology. This final report presents the status of the DACS after the initial 36 months of development. Descriptions of the software engineering computer database and the technology information base are provided. This report also contains information on the types of products developed during this reporting period and the technical inquiries received relating to soft-

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

ware technology.

PREFACE

This final technical report summarizes the accomplishments of the Data & Analysis Center for Software (DACS) covering the period August 1978 to August 1981. This effort was sponsored by the Rome Air Development Center (RADC) Air Force Systems Command, Griffiss Air Force Base, New York, under contract No. F30602-78-C-0255. Mr. John Palaimo of RADC/COEE served as the technical monitor for this effort.

In addition to the authors of this report, project contributions were made by Gineen Brement, Lorraine Duvall, Amy Hahn and Arloene Martin, from IIT Research Institute.

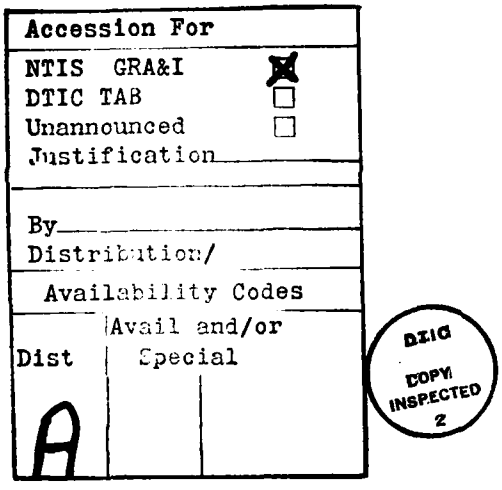


TABLE OF CONTENTS

		Page
I.	INTRODUCTION	1
	1.1 Background	1
	1.2 Objectives of the Center	1
	1.3 Report Contents	2
II.	TASK 1 ESTABLISH AND OPERATE CENTER	4
	2.1 Summary of Technical Progress and Activities	4
	2.1.1 Goals Set for the DACS	4
	2.1.2 Implementation Approach	4
	2.1.3 Summary by Phase	4
III.	TASK 2 DEVELOP SOFTWARE EXPERIENCE DATABASE	7
	3.1 Introduction	7
	3.2 The DACS Software Experience Datasets	7
	3.2.1 Baseline Software Dataset	7
	3.2.2 Software Reliability Dataset	9
	3.2.3 RADC Productivity Dataset	9
	3.2.4 NASA/SEL Dataset	11
	3.2.5 V&V Dataset	13
	3.2.6 Standardized Productivity Dataset	13
	3.3 The Data Acquisition Program	13
	3.3.1 Data Acquisition Approach	16
	3.3.1.1 Identification of Potential Data Sources	16
	3.3.1.2 Identification of Data From Government Funded Programs	16
	3.3.1.3 Identification of Other Sources of Data	18
	3.3.2 Data Identification Follow-up	18
	3.3.3 Sources Identified	19
IV.	TASK 3 DATA ANALYSIS PROGRAM	22
	4.1 Introduction	22
	4.2 The DACS Data Compendium Series	22
	4.3 The DACS Technical Monograph Series	26
	4.3.1 Dataset Comparison	26

TABLE OF CONTENTS Cont'd

		Page
	4.3.2 Analysis of the Effects of MPPs	26
V.	TASK 4 DETERMINE DATA REQUIREMENTS	28
	5.1 Introduction	28
	5.2 Database Evaluation	28
	5.3 Productivity Data Collection Forms	28
	5.4 Conversion Data Collection Forms	29
	5.5 Data Needs for Software Productivity and Reliability	30
VI.	TASK 5 DEVELOP AN INPUT PROCESSING FUNCTION FOR DATA	32
	6.1 Introduction	32
	6.2 Review and Evaluation of Data	32
	6.2.1 Determination of Applicability and Compatibility	34
	6.2.2 Determination of Completeness	37
	6.2.3 Determination of Accuracy and Validity	39
	6.3 Summarization and Reformatting of Data	41
	6.3.1 Summarization Methodologies	41
	6.3.2 Reformatting Methodologies	41
	6.4 Processed Datasets	43
	6.4.1 Standardized Productivity Dataset	43
	6.5 Summary	44
VII.	TASK 6 CURRENT AWARENESS PROGRAM	45
	7.1 Introduction	45
	7.2 DACS Newsletter	45
	7.3 DACS Bulletin	46
	7.4 Information Publications	46
	7.5 Technical Symposia Activities	46
	7.6 User Awareness	52
	7.7 Summary	52
VIII.	TASK 7 THE SCIENTIFIC AND TECHNICAL INFORMATION BASE	54
	8.1 Introduction	54
	8.2 The Software Engineering Document Library	54
	8.2.1 Identification and Acquisition of Documents	55

TABLE OF CONTENTS Cont'd

		Page
	8.2.1.2 Identification of Documents	55
	8.2.1.2 Acquisition of Documents	55
	8.2.1.3 Periodicals	58
	8.2.1.4 Proprietary Information Handling Procedures	58
	8.2.2 Document Processing and Database Update	58
	8.2.3 Retrieval of Bibliographic Information	63
8.3	The SERP Database	63
	8.3.1 Identification of Unpublished Research Project Information	64
	8.3.1.1 Commerce Business Daily	64
	8.3.1.2 Electronic News	65
	8.3.1.3 DTIC Work Unit Summary	65
	8.3.1.4 Software Engineering Notes	65
	8.3.1.5 Computing Newsletter	65
	8.3.1.6 DACS Newsletter	65
	8.3.1.7 Personal Contacts	65
	8.3.2 Special Effort to Identify Non-Government Research Projects	65
	8.3.3 Processing Information for the SERP Database	66
8.4	The DACS Glossary	66
	8.4.1 Coordination with the IEEE Terminology Standard	68
	8.4.1.1 Establishing a Working Group	68
	8.4.2 IEEE and DACS Glossary Interaction	68
8.5	The DACS Thesaurus	70
IX.	TASK 8 PRODUCTS AND SERVICES PERPARATION AND DISTRIBUTION	72
9.1	Introduction	72
9.2	Data Subsets	72
9.3	Data Compendiums	72
9.4	Technical Reports	74
9.5	Newsletter	75
9.6	Informational Database Products	75
9.7	Technical Inquiries	76
9.8	Miscellaneous Services	77

TABLE OF CONTENTS Cont'd

		Page
X.	TASK 11 COST RECOVERY IMPLEMENTATION PLAN	81
	10.1 Cost Recovery	81
	10.1.1 Cost Recovery System Implementation Plan	81
	10.1.2 Alternative Pricing Strategies	81
	10.2 Setting Prices for Products	81
	10.3 DACS Products to be Distributed at No Charge	82
	10.4 DACS Products Priced at Material Cost	82
	10.5 Setting Prices to Recover Total Cost	83
	10.6 Setting Prices to Recover a Percentage of Total Costs	85
	10.7 Charging to Optimize Revenue	86
	10.8 Setting Prices to Reflect Market Value	86
	10.9 Participation Plan	87
	10.10 Exceptions to Pricing Policies for the DACS Products	87
	10.11 Promotional to Current Users	90
	10.11.1 Promotional to Current Users	90
	10.11.2 Promotional to New Users	91
	10.11.2.1 Identification of Demographics of Expected User Population	91
	10.11.2.2 Promoting the DACS in Professional Journals	91
	10.11.2.3 Promoting the DACS Using Free Periodicals	92
	10.11.2.4 Promoting the DACS through Librarians	92
	10.12 Summary of Cost Recovery Plan	93
XI.	SPECIAL TASKS	94
	11.1 Introduction	94
	11.2 Install NBS Software Tools Database	94
	11.2.1 Background	94
	11.2.2 Problem	94
	11.2.2.1 DEC 20	95
	11.2.2.2 Results of Investigation of DEC 20	95
	11.2.2.3 Honeywell 6180 - Multics	95

TABLE OF CONTENTS Cont'd

Page

11.2.2.4 Results of Investigation of Multics . . . 96
11.2.2.5 Conclusions 96
11.3 NSW Tools Training Project 96
11.3.1 Background 96
11.3.2 Problem 97
11.3.3 Summary of Study 97

XII. TASK 9 AND 10 CENTER EVALUATION AND EFFECTIVENESS 98
12.1 Task 9 Center Evaluation Mechanisms 98
12.1.1 Introduction 98
12.1.2 The DACS User Survey 98
12.1.2.1 Introduction 98
12.1.2.2 Summary of Results 98
12.2 Task 12 Improving Center Effectiveness 111
12.2.1 Introduction 111
12.2.2 User Inquiries 111
12.2.2.1 Phase I 111
12.2.2.2 Phase II 113
12.2.2.3 Phase III 116
12.2.3 Information Processing 118

XIII. OBSERVATIONS AND RECOMMENDATIONS 120
13.1 Observations 120
13.2 Recommendations 121

REFERENCES 123

APPENDIX A - PARTIAL LIST OF DACS USERS A-1

APPENDIX B - SUBJECTS OF TECHNICAL INQUIRIES AND BIBLIOGRAPHIC REQUESTS . B-1

APPENDIX C - STATISTICS ON KEYWORD DISTRIBUTION IN THE SEB DATABASE . . . C-1

APPENDIX D - STATISTICS ON KEYWORD DISTRIBUTION IN THE SERP DATABASE . . D-1

LIST OF FIGURES

FIGURE		Page
3-1	LIFE CYCLE PHASES DESCRIBED BY DACS DATASETS	8
3-2	DECOMPOSITION OF THE RADC PRODUCTIVITY DATASET	12
3-3	DACS HAS IMPLEMENTED AN AGGRESSIVE, ORDERLY, SYSTEMATIC DATA SOLITATION PROCEDURE	17
3-4	SOFTWARE DATA AVAILABILITY QUERY	20
4-1	OVERVIEW OF DATA ANALYSIS TASKS	23
6-1	THE INPUT PROCESSING FUNCTION FOR DATA	32
6-2	PROPORTION OF DATASETS CONTAINING A DATA ITEM	40
7-1	NAME, FORMAT AND CONTENT OF THE NEWSLETTER WERE DEVELOPED BY DACS PERSONNEL	47
7-2	THE DACS BULLETIN FOCUSES ON A SINGLE TOPIC OR REPORT	48
8-1	DOCUMENT PROCESSING PROCEDURES	62
8-2	INFORMATION IS COLLECTED FOR ENTRY INTO SERP DATABASE	67
10-1	HYPOTHETICAL PRICE DEMAND CURVE	84
10-2	PARTICIPATION PLAN BROCHURE	88
12-1	DACS USER SURVEY	99
12-2	ORGANIZATIONAL CATEGORIES	102
12-3	JOBS	103
12-4	AREAS OF CONCERN	104
12-5	RATING OF PRODUCTS	105
12-6	CONSULT OTHER SOURCES	106
12-7	HOW WERE YOU INTRODUCED TO THE DACS	108
12-8	BENEFITS OF DACS PRODUCTS AND SERVICES	110

LIST OF FIGURES Cont'd

FIGURE		Page
12-9	DACS ACTIVITIES	112
12-10	INQUIRIES/MONTH PHASE II	114
12-11	INQUIRIES/MONTH PHASE III	117

LIST OF TABLES

TABLE		Page
2-1	CENTER ACTIVITIES BY PHASE	6
3-1	SOFTWARE RELIABILITY DATASET CHARACTERISTICS OF SOFTWARE SYSTEM STUDIED	10
3-2	STANDARDIZED PRODUCTIVITY DATASET DATA DEFINITION	14
3-3	DACS HAS IDENTIFIED USEFUL DATA FROM A VARIETY OF SOURCES	21
6-1	SAMPLE COVERAGE DATA MATRIX	36
6-2	SAMPLE INTEGRATION DATA MATRIX	38
7-1	DACS TECHNICAL PRESENTATIONS AND PAPERS	50
8-1	MEANS OF IDENTIFYING SOFTWARE ENGINEERING DOCUMENTS	56
8-2	DOCUMENT TYPE	57
8-3	BREAKDOWN STATISTICS FOR JOURNAL ARTICLES AND PAPERS AS OF 08/14/81	59
8-4	STEERING COMMITTEE OF THE TERMINOLOGY TASK GROUP	69
8-5	CLUSTER TERMS	71
9-1	DACS PRODUCTS AND SERVICES PRODUCED AND DISTRIBUTED THROUGH 8/1/81	73
9-2	SAMPLE DACS TECHNICAL INQUIRIES	78

LIST OF ACRONYMS

ACM SIGSOFT	Association for Computing Machinery Special Interest Group on Software Engineering
AFLC	Air Force Logistics Command
ANSI	American National Standards Institute
CAI	Computer Aided Instruction
CBD	Commerce Business Daily
CPR	Conversion Problem Report
DACS	Data & Analysis Center for Software
DLA	Defense Logistics Agency
DoD	Department of Defense
DRE	Detailed Resource Expenditure
DTIC	Defense Technical Information Center
EIA	Electronic Industries Association
FI	Forecasting International Ltd.
FY	Fiscal Year
GACIAC	Guidance and Control Information Analysis Center
GAO	General Accounting Office
HOL	Higher Order Language
IAC	Information Analysis Center
IC	Information Center
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IG	Inspector General
IITRI	IIT Research Institute
ISPA	International Society of Parametric Analysts
KWIC	Keyword-in-Context

MCIC	Metals and Ceramics Information Center
MDQS	Management Data Query System
MIPR	Military Interdepartmental Purchase Request
MPP	Modern Programming Practices
MRDS	Multics Relational Data Store
NASA/SEL	National Aeronautics and Space Administration/Software Engineering Laboratory
NBS	National Bureau of Standards
NSW	National Software Works
NTIS	National Technical Information Service
OEM	Original Equipment Manufacturers
O&M	Operations and Maintenance
RAC	Reliability Analysis Center
RADC	Rome Air Development Center
R&D	Research and Development
SEB	Software Engineering Bibliographic
SERP	Software Engineering Research Projects
SET	Software Engineering Terminology
SOA	State-of-the-Art
SPR	Software Problem Report
SSIE	Smithsonian Scientific Information Exchange
SSO	Software System Overview
STI	Scientific and Technical Information
STINFO	Scientific and Technical Information
SVC	Service Request Record
TTG	Terminology Task Group
V&V	Verification and Validation

SECTION I

INTRODUCTION

1.1 Background

The need for data and information about computer software, its development process and the software technology area in general, has been widely recognized by the software community. Comprehensive data and information are needed to provide a better understanding of the factors contributing to the cost, productivity, reliability and quality of software. These data are also needed to develop and evaluate better and more efficient methods of producing software, to predict costs of future developments, to determine the best design, development, and testing methodologies; and to develop ways to effectively measure and predict various software factors, such as quality, reliability, testability, and maintainability. Project managers need data and information from which baselines and guidelines can be developed to assist them in planning, measuring and tracking of software development projects. Furthermore, with the rapid expansion of software engineering technology and the proliferation of tools and techniques, it has been extremely difficult for an individual or organization to remain cognizant of the current status of the software engineering field. This situation has resulted in duplication of efforts in software research and has seriously hampered the transfer of technology from the software research environment to the user sector of the software community.

The Air Force has long recognized the need for an information analysis center to serve the government, industrial, and university community as a focal point for software development and experience data. In 1976 the Rome Air Development Center (RADC) contracted with IIT Research Institute (IITRI) to design a center that would acquire, analyze, synthesize, and disseminate information on software engineering technology [DUVALL-76]. Subsequently, in August of 1978, RADC contracted with IITRI to develop this center, named, The Data & Analysis Center for Software (DACS). It is the purpose of this report to record the activities, accomplishments, and status of the development of the DACS during its 36-month period from August 1978 through August 1981.

The DoD and other Federal Agencies have found that the establishment of special purpose information analysis centers and technology transfer programs is effective in overcoming problems in technology implementation and diffusion of mission-oriented developments. NASA, for example, has been able to demonstrate a benefit-to-cost ratio in excess of ten to one for its Technology Utilization Program. Similarly, the DoD has sucessfully operated Information Analysis Centers for metals, ceramics, hardware reliability, and machinability data [MEYER-79]. The DACS was designated a DoD Information Analysis Center (IAC) in January 1981.

1.2 Objectives of the Center

The DACS provides centralized source for current, and readily-usable

data and information concerning software technology. The objectives of this software information resource are:

- Encourage the diffusion of technology to DoD, civil agencies, government contractors, etc.
- Provide data for software technology research.
- Increase the productivity and quality of computer software by improving the transfer of software engineering technology.
- Assist in diffusing new technology throughout the U.S. industrial base, thereby expanding its capability and competitive posture.
- Provide scientific and technical information analysis services to DoD, civil agencies, government contractors, and the private sector in areas relating to software technology needs, developments, and trends.
- Minimize duplication of effort thereby reducing costs.

1.3 Report Contents

This report provides a summary of the activities of the DACS which were performed with these objectives in mind. It contains thirteen sections and four appendices.

The following is a short description of the topics covered and the specific sections in which they are discussed.

Section I	Background and Objectives of the Center
Section II	Summary of Technical Progress and Activities by Phase
Section III	Descriptions of the Data Acquisition Program and the DACS Software Experience Database
Section IV	Description of the Data Analysis Program
Section V	Description of data parameter definition, data collection forms and related technical reports
Section VI	Description of the Input Processing Functions at the DACS
Section VII	Description of the Current Awareness Program including Newsletters, Bulletins and Technical Presentations
Section VIII	Description of the Scientific and Technical Information Database, the Software Engineering Research Project Database, and the Thesaurus and Glossary

Section IX	Discussions of the Various DACS Products and Services including Data Subsets, Data Compendiums, Technical Monographs, State-of-the-Art Reports, and Bibliographic Searches and Consulting Services
Section X	Discussion of the Proposed Cost Recovery Program
Section XI	Discussion of DACS Special Tasks including a Description of the NBS Software Tools Database and an Evaluation of Tool Training Aids for the NSW Tools Library
Section XII	An Evaluation of Center Effectiveness including a Description of Day-to-Day Operational Data Collected and the Results of a User Survey
Section XIII	Conclusions and Recommendations for Improving Center Effectiveness
Appendix A	Partial List of DACS Users
Appendix B	Subjects of Technical Inquiries and Bibliographic Requests
Appendix C	Statistics on Keyword Distribution in the SEB Database
Appendix D	Statistics on Keyword Distribution in the SERP Database

SECTION II

TASK 1 ESTABLISH AND OPERATE CENTER

2.1 Summary of Technical Progress and Activities

2.1.1 Goals Set for the DACS

The activities of the DACS were oriented toward:

(1) The dissemination of state-of-the-art information on software technology of general interest to the software engineering community.

(2) The development of a Software Engineering Library and Bibliographic Database containing information relating to all aspects of software technology.

(3) The development of a Software Experience Database containing data descriptive of the development and maintenance processes of a variety of software projects, which has been made available to researchers.

(4) The analysis of the data contained in the Software Experience Database.

(5) The conducting of a user awareness program through publication of newsletters and bulletins, presentations at professional seminars, and active participation in professional and technical organizations.

2.1.2 Implementation Approach

These goals were implemented on a build approach basis. That is, an information base was established and used to provide incremental additions of products and services to demonstrate the center's immediate and growing effectiveness. This build approach is highlighted by dividing the descriptions of the progress to date into three phases.

2.1.3 Summary by Phase

Table 2-1 lists the major milestones for each phase of the development of the DACS. Phase I covers the 7 1/2 month period from the initiation of the contract (15 August 1978) to 1 April 1979. This phase included establishing communication channels with the software technology community, generating the procedures (both manual and automated) for developing the technology information base, initiating this development, producing a state-of-the-art report on quantitative software models, and answering 91 informational, technical, and document inquiries.

Phase II (1 April 1979-15 February 1980) included a continuation of

many of the activities performed under Phase I, as illustrated in Table 2-1. In addition to these continued activities, a DACS Glossary, a set of data collection forms and a state-of-the-art report on software maintenance were produced. The total number of inquiries received and processed during this 10-month period was 2,162. These inquiries included requests for technical information, documents, and information on the DACS.

Phase III (15 February 1980-14 August 1981) included a continuation of all activities performed under Phase II. In addition to these continued activities, another DACS Glossary and Thesaurus were produced. A data compendium summarizing the National Aeronautics Space Administration/Software Engineering Laboratory (NASA/SEL) database and a technical monograph describing data analysis results were produced. The technology information base was expanded with increased emphasis on acquiring software experience data through the Data Acquisition Program and the development of a Software Engineering Research Projects Database. Increased emphasis was also placed on the analysis of the data in the DACS Software Experience Database. This resulted in the development of a set of report generation, graphical and data analysis software packages which will be utilized to provide synthesized information to researchers, managers and software developers on request. The total number of inquiries received and processed during this 18-month period was 2120.

By the end of Phase III (which is the end of this contract) the DACS has accomplished the following objectives:

- Provided a centralized source of current literature, research activities and data concerning software development and maintenance technology
- Provided an awareness of the DACS to the software technology community.
- Produced products and provided services that were of immediate use to the community.
- Established a technology information base to be used as a framework for analysis, consolidation, and synthesis.
- Developed a data analysis capability which will be used as a basis for performing software engineering research and consulting services.

The remainder of this report provides discussions on the DACS Software Experience Database, the DACS Technology Information Base, the products produced, and the services rendered.

TABLE 2-1

CENTER ACTIVITIES BY PHASE

Phase I (15 August 1978-1 April 1979) 7½-Month Period	Phase II (1 April 1979-15 February 1980) 10½-Month Period	Phase III (15 February 1980-15 August 1981) 18-Month Period
Established physical facilities. Established mechanisms for processing requests. Initiated document processing.	Refined procedures. Distributed and analyzed the first DACS User Survey.	Continued refining procedures. Designed and implemented an active data acquisition program. Designed a cost recovery program. Distributed and analyzed the second DACS User Survey.
Compiled Newsletter mailing list. (2143 names/organizations)	Updated Newsletter mailing list. (2619 names/organizations)	Updated Newsletter mailing list. (2757 names/organizations)
Prepared and distributed six monthly Newsletters.	Prepared and distributed three quarterly Newsletters and eight Bulletins.	Prepared and distributed six quarterly Newsletters and twelve Bulletins.
Produced the Data Parameters Report.	Produced and distributed a set of productivity data collection forms.	Produced and distributed a set of conversion data collection forms.
Organized the IEEE Terminology Task Group.	Coordinated activities of the IEEE Terminology Task Group.	Continued coordinating activities of the IEEE Terminology Task Group.
Prepared and presented introductory papers on the DACS	Prepared and presented papers on the DACS, quantitative software models, database evaluation, and software reliability.	Prepared and presented papers on DACS data analysis program, approaches to technology transfer and data needs for software engineering.
Initiated the acquisition of the NASA/SEL dataset	Installed the NASA/SEL and Musa dataset on the RADC computer. Produced a data compendium based on the NASA/SEL dataset. Distributed copies of the RADC Productivity and the Musa Reliability Datasets.	Installed, partially analyzed and distributed updated NASA/SEL dataset. Installed and partially analyzed V&V dataset. Produced a second data compendium.
Designed and implemented the datafiles for the Glossary, Thesaurus, and bibliographic databases. Generated computer queries. Generated Thesaurus.	Produced and distributed the DACS Glossary and the Thesaurus. Produced custom bibliographic searches. Produced and distributed the users guide, "Bibliographic Services, Custom Searches" (BIB-I)	Continued maintianing technology information base. Prepared two comprehensive software engineering annotated bibliographies. Produced the second DACS Glossary and an updated Thesaurus Performed engineering research, distilled and synthesized bibliographic database information.
Produced SOA Report "Quantitative Software Models" (SRR-1).	Produced SOA Report "Software Maintenance Technology."	Produced a software engineering research projects directory and one technical monograph Developed a data analysis plan.
Processed 95 technical and document inquiries.	Processed (as of 1 February) 281 technical inquiries, and 1212 document inquiries.	Processed as of 15 August 1981 an additional 551 technical inquiries and 3139 document inquiries.

SECTION III

TASK 2 DEVELOP SOFTWARE EXPERIENCE DATABASE

3.1 Introduction

This section of the report contains descriptions of the DACS Software Experience Database and the data acquisition program.

The DACS computer database presently consists of five sets of data distinguishable by data source, data collection and acquisition methodology, life cycle phase represented and data parameters present. These datasets have been implemented on the RADC HIS 6180 computer system using the Management Data Query System (MDQS) to facilitate data retrieval and analysis. Each of these five sets of data are described in subsequent sections of this report and are termed:

- Baseline Software Dataset
- RADC Productivity Dataset
- Software Reliability Dataset
- NASA/SEL Dataset
- Verification and Validation (V&V) Dataset

These datasets contain data from various life cycle phases. The most overlap appears in the data relating to the system test and evaluation phase. This is illustrated in Figure 3-1.

3.2 The DACS Software Experience Datasets

3.2.1 Baseline Software Dataset

This DACS dataset consists of data describing software problem reports acquired by RADC from six large software development efforts. These projects and the data are described in [RYE-77], [BAKER-77], [WILLIAM-77], [FRIES-77], [THAY-76], and [HECT-77].

The system application areas encompass command and control, real-time control for land-based radar, onboard guidance and navigation, and database management. The majority of the datasets contain the date the software problem was opened and closed, the module that manifested the problem, the module that was changed to correct the problem, the problem category and priority, and the correction type.

Three of the datasets (data related to each system is also called a dataset) contain module descriptive information which designates the number of source instructions, the programing language, the type of construction, and a functional area designation. One dataset records information on each

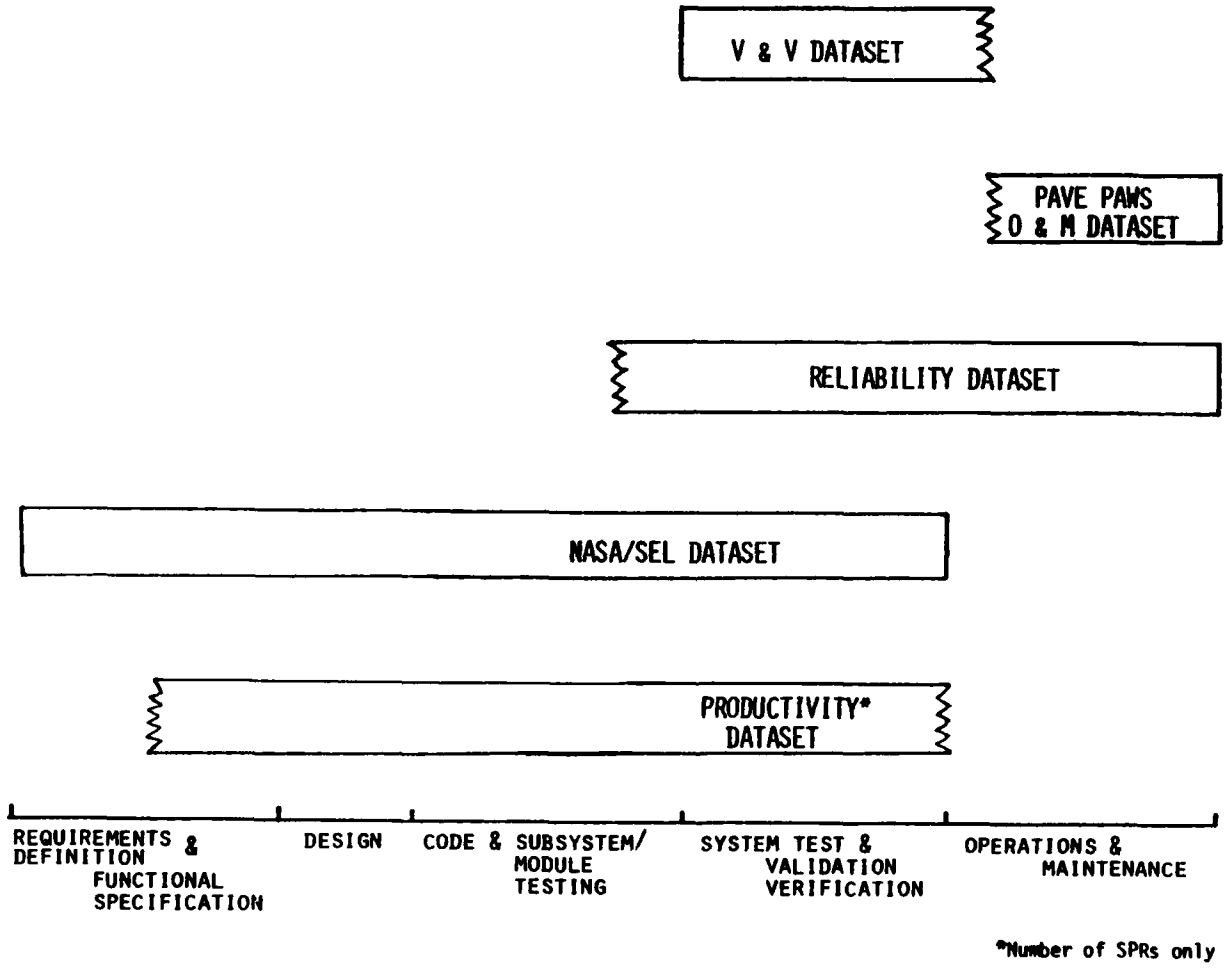


FIGURE 3-1

LIFE CYCLE PHASES DESCRIBED BY DACS DATASETS

test run.

This dataset presently consists of 26,594 Software Problem Report records, 2719 Run Analysis Report records, and 2591 Module Description records. [DUV-79I] and [DUV-79II] describe the implementation of these datasets on the RADC computer system.

3.2.2 Software Reliability Dataset

This dataset was provided to the DACS by John D. Musa, Bell Telephone Laboratories, Whippany, NJ, with the hope that it would be of general benefit for software reliability researchers and that it would permit independent verification of his conclusions on the execution time theory of software reliability [MUSA-79].

The primary objective for collecting this data on 16 software systems was to assist software managers in monitoring status and predicting schedules. Careful controls were employed to ensure that the data would be of high quality.

Failure interval data (failure number, failure interval, day of failure) is provided for all 16 systems with a total sample size of 2831. Failure correction dates and resource expenditure information is available for four of the systems.

The size of the software systems varies from approximately 6,000 to hundreds of thousands of delivered object instructions. The nature of the systems are real-time command and control, real time commercial, operating systems, a time sharing system, and a word processing system. Operational failure data is available from 11 of the systems, system test failure data from eight of the systems, and subsystem test failure data from one of the systems. Four of these datasets contain failure and resource expenditure data for both the system test and operational phases of the software life cycle. The characteristics of these software systems are summarized in Table 3-1.

3.2.3 RADC Productivity Dataset

This dataset was compiled by Richard Nelson of RADC and contains summary information from over 400 software projects, some dating back to the early sixties, mostly on DoD applications. The dataset incorporates productivity data, error data, project duration, total effort, language data, and information on the usage of various software implementation technologies. At present, 93% of the projects contain productivity data, 7% contain error data, 91% contain language usage data and about 50% contain information on implementation technologies. In total, this dataset covers approximately 21 million lines of code. The data has been collected from a large variety of sources, both public and private.

TABLE 3-1

SOFTWARE RELIABILITY DATASET

CHARACTERISTICS OF SOFTWARE SYSTEMS STUDIED

System No.	Phases for Which Data Available	Delivered Object Instructions	Programmers	Size of Failure Sample	Nature of System
1	S,O	21,700	9	136	real time command and control
2	S,O	27,700	5	54	"
3	S,O	23,400	6	38	"
4	S,O	33,500	7	53	"
5	S*	2,445,000	275	831	real time commercial
6	SS	5,700	8	73	commercial (subsystem)
14C	O*	Hundreds of thousands	110	36	real time
17	S	61,900	8	38	military
27	S	126,100	8	41	military
40	S	180,000	8	101	military
SS1A	O*	Hundreds of thousands	not known	112	operating system
SS1B	O*	"	"	375	"
SS1C	O*	"	"	277	"
SS2	O*	"	"	192	time sharing system
SS3	O*	"	"	278	word processing system
SS4	O*	"	"	196	operating system

Phases:
SS = subsystem test
S = system test
O = operational
Data is for a complete phase unless starred.

A preliminary analysis and a description of this dataset is contained in [NELSON-78] .

This dataset is represented in Figure 3-2 by a hierarchical scheme which decomposes the dataset based on the presence or absence of the following data parameters:

- Production/Error Data
- Programming Language
- Implementation Techniques
- Design Languages

Each node of the hierarchy contains (1) the number of data elements meeting the selection criteria at the node (contained in parenthesis in the figure) and (2) a summary statistic (median productivity or median error rate) based upon the data meeting the node's selection criteria. Since the representation is a hierarchy, the sum of all of the nodes of a family contain the same number of data elements as the node that is the parent of the family. The root (INPUT DATA) of the hierarchy contains 407 projects, the total number of projects in the database. The first selection criteria is whether the input data is classified ERROR DATA or PRODUCTIVITY DATA. Proceeding down the ERROR DATA leaf of the hierarchy, 30 records contain ERROR DATA, 377 contain NO ERROR DATA. For the 30 records in the Database that contain error data, the MEDIAN ERROR RATE is 0.01 Software Problem Report (SPR) per line of delivered source code. The next selection criteria is IMPLEMENTATION TECHNIQUE. Fourteen of the 30 projects contained NO DATA, 6 of the 30 projects used MODERN PROGRAMMING PRACTICES (with a median error rate of 0.005), and 10 of the 30 projects used TRADITIONAL PROGRAMMING PRACTICES (with a median error rate of 0.01).

The PRODUCTIVITY leaf of the hierarchy is interpreted in a similar manner to that described above. For example, of the 379 projects containing PRODUCTIVITY DATA, there is a MEDIAN PRODUCTIVITY RATE of 223 lines of source code delivered per man-month.

3.2.4 NASA/SEL Dataset

The Software Engineering Laboratory (SEL), at NASA Goddard Space Flight Center, was organized in August 1976 to monitor existing software methodologies and to develop and measure the effectiveness of alternative methodologies [BASILI-79]. To accomplish these objectives, the SEL has been collecting data during the development of their software products and has submitted this data to the DACS for subsequent analysis and distribution. This dataset will continue to be updated as more data is collected.

As of 1 December 1980, the NASA/SEL Dataset at the DACS contained information on 20 software development projects. These projects include

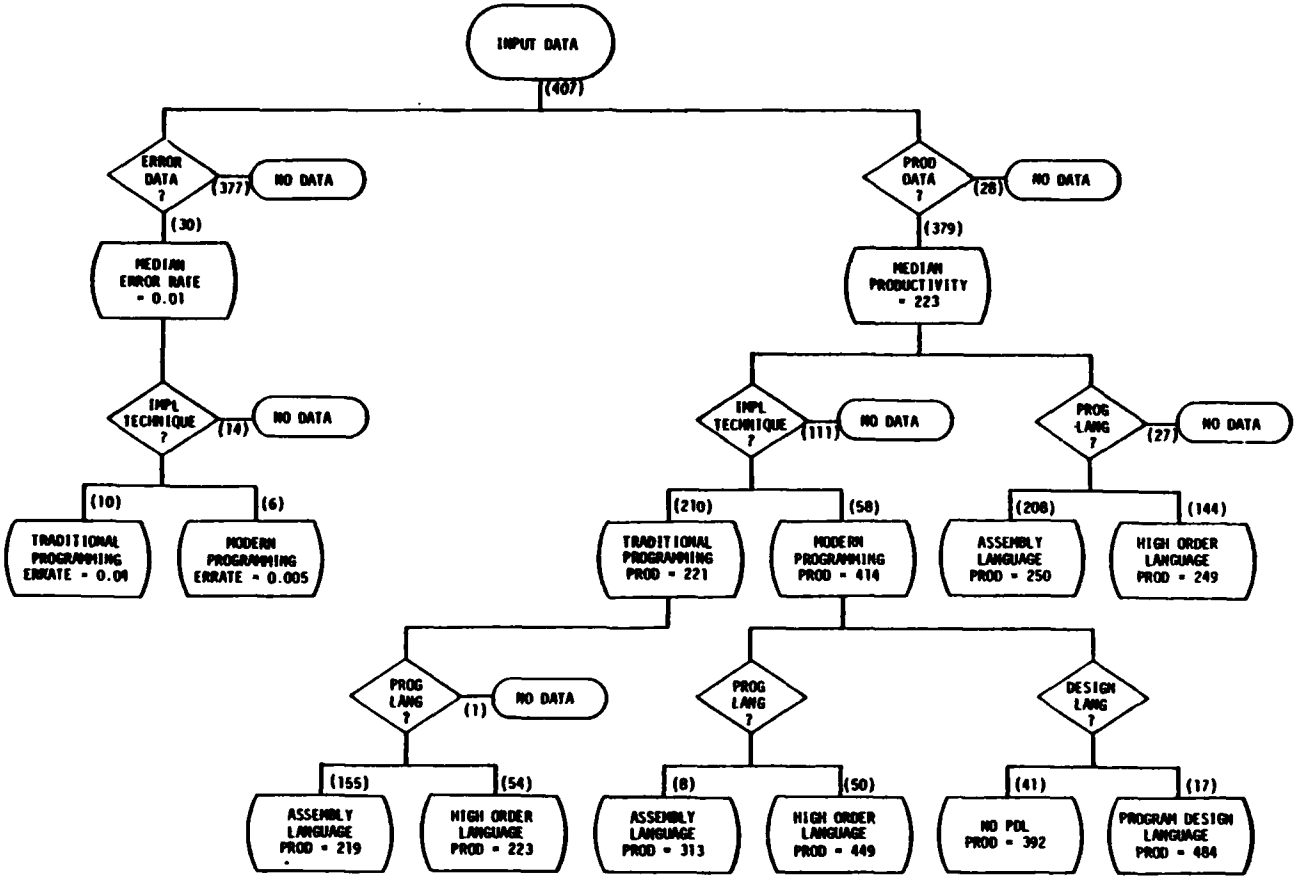


FIGURE 3-2

DECOMPOSITION OF THE RADC PRODUCTIVITY DATASET

orbit and attitude determination systems, mission support packages, attitude ground support systems, a human resources scheduling system, a FORTRAN source analyzer, and other software support systems. The computer programs were written in FORTRAN or a combination of FORTRAN and Assembly language for a PDP 11/70 or a System 360.

The majority of the data is from component status reports (22,000 records), followed closely by run analysis report data (15,000 records). The remainder is project comment information (2994 records), change reports (3135 records), resource summary reports (1700 records), and component summary reports (over 1000 records).

The datasets are not complete for all projects because some were well underway before the data collection effort was initiated and some projects are still undergoing development.

The data supplied by SEL has been summarized in a data compendium produced by the DACS [TURN-81A] and described in Section 4.6. This dataset has also been compared with the RADC Productivity Dataset. Results of the comparison are documented in [TURN-81B].

3.2.5 V&V Dataset

This dataset contains data collected during the independent V&V of five software projects. Data on three of these projects has been recorded at the subsystem level as these subsystems underwent separate V&V. The purpose of this data collection effort was to record the types of errors which can be detected during independent V&V activities.

The dataset consists of general project development background information and a number of anomoly reports on each project. General background data includes; size, development duration, phase dates, software application, and language. Each anomoly encountered during V&V was recorded on an anomoly report. Each report consists of data specifying the anomoly location, category, severity, tools used and the effect of the anomoly. Over 1500 anomoly reports are present in the dataset.

3.2.6 Standardized Productivity Dataset

Common elements of the RADC, NASA, and V&V datasets have been combined into one dataset using a standardized record format which is expandable as new data parameters are identified as being needed for quantitative analysis studies. This combined dataset contains those parameters which have been identified by DACS analysis personnel as most common across the three datasets. The parameters selected for inclusion in this dataset are defined in Table 3-2.

This combined dataset is called the Standardized Productivity Dataset. It is used extensively as input to the DACS data analysis program.

3.3 The Data Acquisition Program

TABLE 3-2

STANDARDIZED PRODUCTIVITY DATASET: DATA DEFINITION

Parameter	Columns	Format	Description
PROJECT CODE	1-4	I(4)	Four digit internally assigned project identifier
PROJECT SIZE	5-11	I(7)	Total delivered source lines for the project (not including unmodified reused code) 0 = data not recorded
PROJECT EFFORT	12-16	I(5)	Total project development effort in manmonths, includes management and clerical 0,-1 = data not recorded
PROJECT DURATION	17-19	I(3)	Total project development time in months 0,-1 = data not recorded
AVERAGE STAFF SIZE	20-22	I(3)	Average number of people assigned to the development of the project 0,-1 = data not recorded
PRODUCTIVITY	23-27	I(5)	Average code production rate during development in lines per manmonth 0,-1 = data not recorded
PROBLEM REPORTS	28-32	I(5)	Number of documented changes, i.e., Problem Reports, Change Requests, to the project; corresponds to the number of errors encountered during development 0,-1 = data not recorded
PDL RATE	33-35	I(3)	Percent of project (based on code) which was developed using a Program Design Language 0 = none used 999 = no data on use
STRUCTURED CODE RATE	36-38	I(3)	Percent of code which is structured 0 = none 999 = no data on use

TABLE 3-2 Cont'd

TOP-DOWN RATE	39-41	I(3)	Percent of code developed in a top-down fashion 0 = none 999 = no data on use
CHIEF PROGRAMMER TEAM RATE	42-44	I(3)	Percent of code developed under a chief programmer team concept 0 = none 999 = no data on use
PROGRAMMER LIBRARIAN RATE	45-47	I(3)	Percent of code developed using a programmer librarian 0 = none 999 = no data on use
CODE REVIEW RATE	48-50	I(3)	Percent of code reviewed under formal procedures 0 = none 999 = no data on use
PRIMARY LANGUAGE	51-61	A(11)	Character string indicating primary language used during development blank = no information
DATASET ORIGIN CODE	62	A(1)	Letter indicating the dataset from which the project data originates R = Nelson Productivity N = NASA/SEL V = V & V Dataset

3.3.1 Data Acquisition Approach

The means used to develop an aggressive data acquisition program are presented in the following subsections and consist of:

- Identifying data sources and acquiring relevant data
- Establishing procedures for automatic submission of data

3.3.1.1 Identification of Potential Data Sources

The DACS has aggressively sought data that has been, or is to be collected on various phases of the software life cycle. This data has been sought from sources in government and industry. Government sources include software development contractors, V&V contractors, and software development and maintenance sources in DoD, NASA and other government agencies.

Both published and unpublished data were sought. For the most part, acquisition of published data was relatively straightforward and routine. This type of information appeared in readily available forms such as published government-funded Research and Development (R&D) reports (Defense Technical Information Center (DTIC) Technical Abstract Bulletins' and National Technical Information Service (NTIS) 'Government Report Abstracts'), technical society and trade journals, conference and symposia proceedings, etc. Items were identified and evaluated for relevancy to the DACS program. DACS has also remained cognizant of the funding of new R&D efforts and other unique publications and conferences.

3.3.1.2 Identification of Data From Government Funded Programs

The major engineering effort in data solicitation has been directed toward unpublished sources, particularly software and systems development and procurement programs, as well as operations and maintenance experience. Data from these sources are less defined and more difficult to identify and access, but have the greatest utility as a DACS resource. Significant strides have been made in the past year to identify such data. Given that data of interest to the DACS may accumulate over a period of several years as a software system moves through the various phases of its life cycle, it is important that such systems be identified at the earliest possible time in their life cycle. Early identification of such programs increases the chances that data can be acquired as development proceeds, thereby tending to decrease the costs of data collection while tending to increase the accuracy and validity of the collected data.

The DACS identifies and establishes interface with software development programs as early as possible in order to achieve a complete and worthwhile history file on those systems tracked. A methodology has been implemented using regularly published official or quasi-official government listings of contract solicitations and awards for identification of programs. The general procedures followed in this approach are depicted in Figure 3-3.

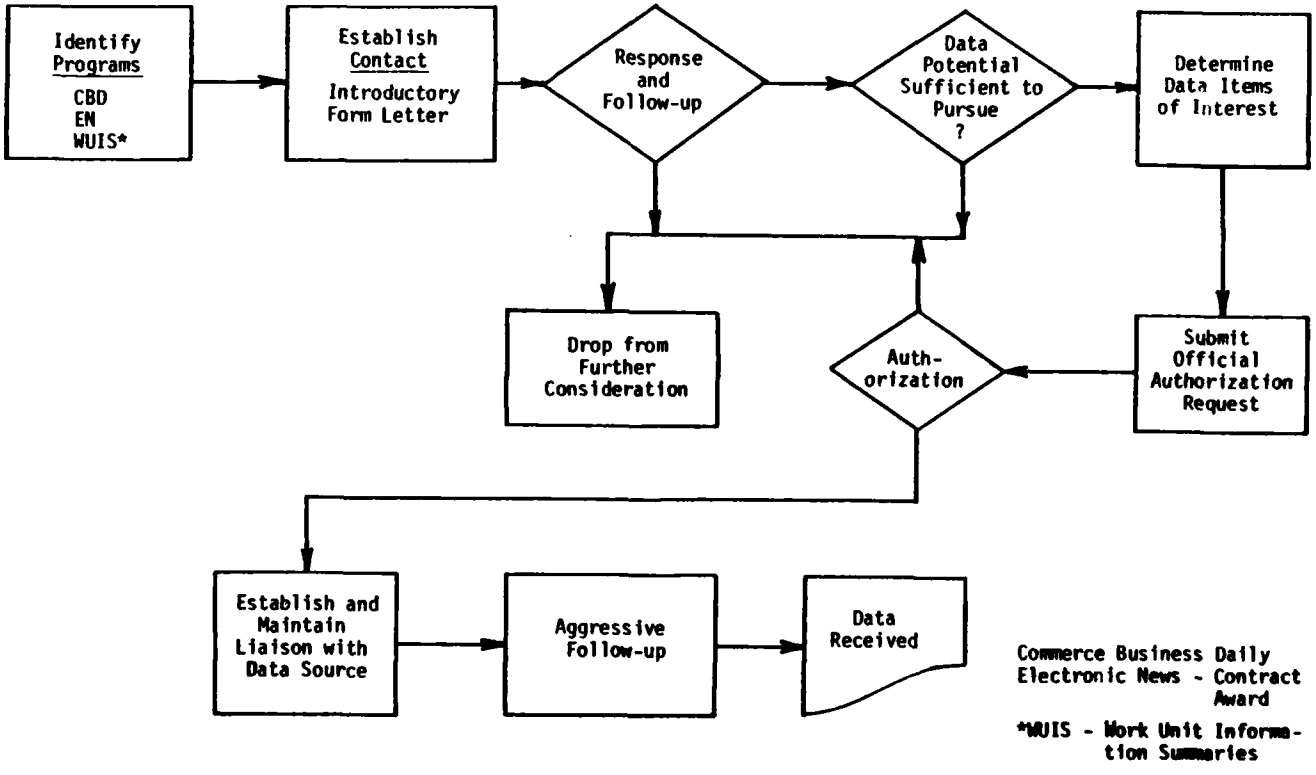


FIGURE 3-3

DACS HAS IMPLEMENTED AN AGGRESSIVE, ORDERLY, SYSTEMATIC DATA SOLICITATION PROCEDURE

Three major information sources are reviewed regularly, namely:

Commerce Business Daily (CBD)

Electronic News (Contract Awards)

DTIC (Work Unit Information System)

The CBD is desirable, as it makes possible identification prior to contract award and enables the DACS to be written into the contract as a recipient of selected data items.

The Electronic News contract awards section is reviewed because it identifies subcontract awards as well as contract awards. It often happens that the CBD announcement does not indicate that software is contained in a system procurement and such software development is often subcontracted. Both the fact that 1) the software will be developed for such a system and 2) the subcontractor who will develop it can be ascertained from Electronic News. When projects likely to produce data of interest to the DACS are identified, solicitation of data follows the same general pattern established for sources identified via the CBD above.

A third source of information which is used to identify R&D and also possible data to the DACS, is the DTIC work Unit Information System. This system identifies programs which are new or on-going that can submit documents to the DACS at the time they are published, thus making the results of significant research available to the DACS and its user community at the earliest possible time.

Most of the data sources who have been willing to contribute data to the DACS fall into the categories of government agencies or government contractors. Another very important source of data is the commercial computer equipment/software manufacturers, as discussed in the next section.

3.3.1.3 Identification of Other Sources of Data

One of the more effective means of acquiring information about data sources is through personal contacts made through interaction with DACS users, work on professional committees (i.e. IEEE or ACM) and attendance and presentations at conferences, workshops, and symposia. These contacts tend to produce leads to data which are not publicized elsewhere and are often the product of internal R&D work, production of commercial software, or research at universities.

3.3.2 Data Identification Follow-up

After a program is identified, the next action is to establish contact and determine the extent of applicability of software data that may emanate from the program. This has been accomplished through use of form letters to contracting agencies and/or questionnaire forms which are tailored to the particular situation. For example, the Software Data Availability Query

Form (Figure 3-4) is sent to a respondee who has indicated that data will be collected. Its purpose is to obtain information on the status, type of software to be developed, and the nature of the data that is expected to be collected.

After the initial response has been received, appropriate follow-up activities are carried out depending on the availability, relevance, and probable quality of the data expected to emanate from the program, as depicted in Figure 3-3.

Follow-up activities include:

- Additional letters
- Telephone calls
- Discussions at conferences/meetings
- Visits to organizations

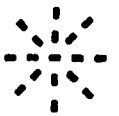
3.3.3 Sources Identified

From the initiation of the aggressive phase of the DACS data acquisition program in March 1980 through 14 August 1981, a total of 699 inquiries were made with the intention of ascertaining the availability of software data. The majority of these inquiries were made by mailing letters as a result of announcements in the CBD. The DACS received 162 responses to its letter inquiries. Of the 162 responses, 66 were evaluated as not worth follow-up for reasons such as software not being developed, project cancellation, or the agency's declining to reveal information about the project. Of the 162 responses to its data inquiry letters, 96 were evaluated as being worth further follow-up activities. Such activities generally involved a request for additional information which took the form of another letter, a telephone call or a visit to the organization. In all, 194 follow-up letters or phone calls and three visits to potential data contributors were made during this time period.

The DACS has distributed a series of data collection forms. The forms were designed by DACS personnel in response to requests from many of the individuals who were approached as part of our data acquisition activities. The objective was to make data collection as easy and fast as is consistent with the concerns of accuracy. These forms are designed to be completed at various stages of the software life cycle. The forms can be used to collect data which will support cross comparisons with data already in the DACS databases and can also be used to collect data to support identified research aims of the DACS for which data has not yet been available. The supplying of these forms to organizations collecting data, facilitates the submission of data which support the needs of the DACS and makes it easier to induce organizations to submit data on a continuing basis. Table 3-3 contains a representative sample of the activities DACS has conducted during its administration of the Data Acquisition Program and the results of those activities.

DACS RADC/COEI
Griffiss AFB, NY 13441
315/330-0937 Autovon 587-3395
Data & Analysis Center for Software

SOFTWARE DATA AVAILABILITY QUERY



Present System Status ☐ Planned ☐ Early Development ☐ Final Development ☐ Operational

Project Name: _____

Description: _____

_____

Number of Systems: _____

Number of Functional Groups: _____

Number of Modules: _____

Number of Lines of Code: _____ Source _____ Object

Contact: _____

Developer: _____

Project Duration: _____

Staff Size: _____ Contract: _____

Management Organization: _____

Cost Reporting Standards: _____

Technical Standards:

_____ _____ _____

_____ _____ _____

Phase of Software Life Cycle for which Data was/will be Collected:	Format of Data: Machine Readable	Format of Data: Hard Copy
( ) Requirements	( )	( )
( ) Specifications	( )	( )
( ) Design	( )	( )
( ) Coding/Implementation	( )	( )
( ) Testing	( )	( )
( ) Operations & Maintenance	( )	( )

Description of Dataset Data Type(s):

( ) Environment
( ) Technology
( ) Productivity
( ) Characteristics
( ) Resource Utilization
( ) Testing
( ) Error/Fault
( ) Change
( ) Other _____

Computers:

	Hardware	Oper. System	Access
Development	_____	_____	_____
Target	_____	_____	_____

Support Software Developed by Project:

_____

_____

Who should be contacted for authorization to release data?

Name: _____ Address: _____

Title: _____ _____

Telephone: _____ _____

Who should be contacted for further information/data on this project?

Name: _____ Address: _____

Title: _____ _____

Telephone: _____ _____

FIGURE 3-4

SOFTWARE DATA AVAILABILITY QUERY

TABLE 3-3

DACS HAS IDENTIFIED USEFUL DATA FROM A VARIETY OF SOURCES

DATA SOURCE	IDENTIFICATION OF SOURCE MODE OF INQUIRY	CONTACT(S)	DESCRIPTION OF PROJECT/DATA EVALUATION OF DATA	RESULTS OF ACQUISITION ACTIVITIES
Westinghouse Electric Company, Pittsburgh, PA	Contact at Westinghouse responded to request for data published in DACS Newsletter Several phone calls visit to Westinghouse	Lee Shaw, Westinghouse	Problem reports from test and integration phase of development of three nuclear reactor software projects Data not of interest to DACS because (1) Problem reports did not contain fault type or enough information to classify faults (2) Data was in hard copy form and several years old, development personnel not available to clarify inconsistencies and/or supply missing information	No data acquired
Software Engineering Laboratory (SEL) U. of Maryland/NASA Goddard	Contact made through attendance at workshop phone calls, letters, visits to SEL	Frank McGarry, NASA Goddard Space Flight Center, Victor Basili and Marvin Zelkowitz, U. of Maryland	Life Cycle Data on 42 projects developed or currently being developed at NASA, Goddard Data is of excellent quality, being collected on continuing basis, already in machine-readable form	Data acquired on magnetic tape with periodic updates supplied by SEL
Logicon, Inc. San Pedro, CA	Specification in RADC contract that data be supplied to DACS phone call to discuss tape specifications	Jane Radatz, Logicon	Compilation, classification of faults detected by independent validation and verification of software Quality data of high interest to DACS	Data delivered to DACS on magnetic tape as per contract
NASA-Ames Research Center Moffett Field, CA and AFWAL/FIGLA Wright-Patterson AFB, OH	Commerce Business Daily form letter, several telephone calls	Dr. Pio deFeo NASA-Ames; Philip Chandler, AFWAL/FIGLA	The data is being collect during a joint three-part effort funded by NASA-Ames and AFWAL/FIGLA. NASA-Ames is acquiring a digital flight control system, during development fault data will be collected to provide baselines AFWAL/FIGLA is acquiring a similar system. When development is complete, faults will be seeded into the development system in accordance with the baselines developed by NASA-Ames. Automated V & V tools will be applied sequentially to the seeded software to determine cost/benefit ratios for each tool. Data will be extremely valuable to DACS.	Data from all phases of the three-part project will be given to the DACS. Format will be consistent with DSDS. Expected delivery, late fall 1981.
Naval Data Automation Command, Washington, DC	Data discussed in technical report processed for Library letter, telephone	Elmer Slaughter, NAVDAC	Detailed records of manpower and computer utilization by type of activity during conversion of at least nine software conversions at NAVDAC sites Good data which should be acquired. DACS has no other conversion data, but has received multiple requests for such data	Negotiations in process-requires funding of $2,500 to transform data to format usable by DACS
IITRI/ECAC Annapolis, MD	Visit by contact to DACS Discussion at DACS, telephone contacts, letter, visit to ECAC	John Jones, IITRI/ECAC	Detailed life cycle change data on two large EMC modelling/analysis software systems. Data collected as part of strict configuration control procedures. Data items include nature of change (i.e., enhancement or correction), resulting increase in size of software, manhours to effect change DACS has no maintenance data, this is best we have thus far seen, should be acquired	DACS will fund costs of transforming data from hard copy to machine-readable form. IITRI/ECAC will do preliminary analysis of data. Data and results of analysis will be delivered August, 1981.
U.S. Air Force Wright Avionics Laboratory, Wright-Patterson AFB, OH	Referral made by IITRI personnel working on another program Several telephone calls, letter, discussion at two workshops attended by both DACS personnel and Dan Ferens	Daniel Ferens, AFWAL/AAAF-2	Data on support costs for AF avionics software. AFWAL is funding a two-part study to determine feasibility of generating a predictive software support cost model. Both stages of study involve data collection Data is directly applicable to research aims of DACS. Should be acquired.	Received data from first stage of study in hard copy form. Will receive data from second stage of study in computer-readable form which will also include data collected during first stage

SECTION IV

TASK 3 DATA ANALYSIS PROGRAM

4.1 Introduction

The DACS data analysis program was aimed at identifying trends, baselines and guidelines; evaluating and refining mathematical models; and evaluating software development methodologies and techniques. Emphasis was directed toward the evaluation of software experience datasets to determine their applicability toward supporting research activities in the areas of software productivity, error rate and the usefulness of Modern Programming Practices (MPPs).

The purpose of this section of the report is to provide a summary description of the specific data analysis tasks performed by the DACS during the time frame covered by this contract. Figure 4-1 depicts a summary of the problem, the processes and products addressed and performed by the DACS during the three year time frame to meet the specific objectives of the data analysis task. The following topics are addressed in the remainder of this section: The DACS Data Compendium Series, and the DACS Technical Monograph Series.

4.2 The DACS Data Compendium Series

Two of the functions of the DACS' data acquisition and analysis program are to provide summaries of the DACS Software Experience Database and subsets of this database. These summaries are designed to provide users of the DACS with useful information regarding the data available through the DACS, and also to provide information regarding the types and depths of analysis activities which may be performed using this data.

The DACS produced two data compendiums under the current contract both of which summarized the data contained in the NASA/SEL Dataset. Only the second version was released; the addition of a large volume of data to this dataset, soon after production of the first compendium, made the first compendium outdated. This compendium summarized the data contained in the dataset by type of data, i.e., general development effort and schedule data, resource expenditure data, and change report data. [BASILI-79] and [TURN-81A] contain a full description of the NASA/SEL data collection methodology as well as summaries by type of data.

The following subsets of the NASA/SEL data have been extracted from the NASA/SEL dataset and put into graphical and tabular form:

General Project Information

Delivered Source Lines of Code
New Lines of Delivered Code
Number of Components
Number of Modules

THE PROBLEMS	THE PROCESSES	THE PRODUCTS
We don't know factors which have the most significant impact in software life cycle cost and performance	Perform literature review to determine state-of-the-art in software life cycle cost and performance estimation	Software experience data summaries (compendiums) describing datasets available from the DACS
	Define the data analysis requirements	Publications containing sets (by category) of tabular data extracted from the DACS database
	Define baseline software experience database format	
	Acquire software experience data on development, maintenance and conversion	Technical reports describing the quality, consistency and suitability of the datasets for analysis
	Develop a library of software storage and retrieval procedures to manipulate baseline data and store in tabular form for subsequent input to statistical analysis programs	Technical reports describing the results of productivity and reliability modeling studies using statistical analysis
		Technical reports describing the suitability, accuracy, robustness, ease of use of quantitative software prediction models for cost, reliability, maintainability prediction

FIGURE 4-1

OVERVIEW OF DATA ANALYSIS TASKS

Number of New Modules
Number of Pages of Documentation
Months of Development Time
Manmonths of Development Effort
Number of Computer Runs
Number of Computer Hours
Number of Program Changes

Project Scheduling Information

Design Start and End Dates
Code and Test Start and End Dates
System Test Start and End Dates
Acceptance Test Start and End Dates
Cleanup Start and End Dates

Code Production History

End of Week Date
Cumulative Number of Source Lines Developed
Cumulative Number of Components Developed
Cumulative Number of Code Changes

Development Effort by Activity, Manhours

Design: Create; Read; Review
Development: Code; Read; Review
Testing: Module; Integration; Review

Profile of Run Purposes and Results

Profile of Types of Changes and Errors

Chronological Failure Data

Failure Number
Date Detected
Date Corrected
Type of Failure or Change
Origin of Failure
Number of Modules Affected
Calender Days Since Last Failure/Change

Data is incomplete for a number of projects. This is because some projects were already under development when the data collection process was initiated or are still in their initial stages of development. Data is missing, most notably, in the computer run analysis area and the error analysis area.

[TURN-81A] contains a set of three tables of summary data which summarize the three major categories of data available across all projects. These data categories are:

- Project Development Data
- Change/Error Data
- Development Methodology Data

Each of these major data categories has been further subdivided into subcategories representing phasing and scheduling data; human and machine resources; project size, composition and development history; run purposes and outcomes; and finally, the distribution of when errors were introduced into the software, as well as the effort required for their correction. These data are available through the DACS.

The number of projects for which data is available in these specific categories is as follows:

Project History and Development Data

The number of projects for which data is available for each subcategory is as follows:

I.	Project Size	19
II.	Development Time	18
	Development Effort	18
III.	Development Time by Phase/Development Effort by Phase	
	Design	24/22
	Code and Test	23/21
	System Test	22/20
	Acceptance Test	21/0
	Cleanup	18/0
IV.	Computer Resource Expenditures	14

Change Report Data

Data is relatively complete for ten of the projects. The number of projects for which data is available for each subcategory is as follows:

I.	Number of Change Reports	21
II.	Distribution of Changes by Phase	10
III.	Distribution of Why Changes Made	13
IV.	Distribution of When Error Entered	18
V.	Distribution of Effort to Resolve Change	20

Development Methodology Data

Data is relatively complete for 21 projects. Software development constraints, as well as the MPPs, techniques and tools utilized during the development of the NASA software, are listed in [TURN-81A].

4.3 The DACS Technical Monograph Series

In January 1981, the DACS initiated a technical monograph series that describes analysis activities that have been performed by the DACS. During Phase III of this contract effort, one technical monograph entitled "A Comparison of RADC and NASA/SEL Software Development Data" was produced [TURN-81B]. Another technical monograph is currently being developed entitled "Effect of MPPs on Programmer Productivity and Software Error Rate."

These technical studies are discussed in the remainder of this subsection.

4.3.1 Dataset Comparison

A form of data analysis for which the DACS is particularly well qualified is the comparison of newly acquired datasets with the datasets already in the DACS Software Experience Database. To establish this capability, FORTRAN programs were developed which retrieve specific sets of data items from any two datasets, perform a statistical analysis, and plot the results on the Anderson Jacobson 832 Terminal. Data retrieval is performed by special purpose Management Data Query System (MDQS) programs adapted specifically for each dataset. The set of FORTRAN programs interface with the International Mathematical and Statistical Library (IMSL) and the Anderson Jacobson graphics support software package.

In December 1980, the DACS acquired the NASA/SEL Dataset consisting of software development data on 20 projects; 18 projects which contained productivity data and 21 projects which contained change report data. This dataset was discussed briefly in Subsections 3.2.4 and 4.2 of this report. This dataset was combined with the RADC Productivity Dataset, resulting in the establishment of the DACS Standardized Productivity Dataset with each data record for each project in a standardized format, suitable for comparative analysis.

In May of 1981, the DACS produced a technical monograph [TURN-81B] which examined several relationships, when the NASA/SEL Dataset was merged with the RADC Productivity Dataset.

The study indicated that the seven relationships examined were not significantly affected by the combination of the two datasets.

4.3.2 Analysis of the Effects of MPPs

A study began in early August 1981 to examine the effects of the use of MPPs with respect to programmer productivity and error rate of software

projects recorded in the RADC productivity, NASA/SEL, and V&V Datasets. The metrics, productivity and error rate, were selected because they are not extremely variable with respect to project size. That is, these metrics are not as sensitive to project size as project duration, effort, and total number of errors.

Programmer productivity and error rate were examined in previous studies as a function of program size, conventional development techniques as opposed to the use of MPPs and higher order language versus Assembly level language usage. These studies did not utilize formal statistical techniques other than linear regression analysis, to compare the effects of program size, language and development methodology on programmer productivity, error rate, duration, et al.

This study addresses the following questions:

(1) Is productivity related to the extent that MPPs are utilized during software development?

(2) If so, what specific MPPs affect productivity most significantly?

(3) What additional data is needed to determine with more confidence if the use of MPPs affect productivity?

(4) Is spatial error rate related to the extent that MPPs are utilized during software development?

(5) If so, what specific MPPs affect spatial error rate most significantly?

(6) What additional data is needed to determine with more confidence if the use of MPPs affect spatial error rate?

The results were not completed during this contract period but will be available at a later date.

SECTION V

TASK 4 DETERMINE DATA REQUIREMENT

5.1 Introduction

Five major activities were performed under this task definition and are discussed in this section. A database evaluation methodology was developed and used to determine the relevance and limitations of the Baseline Software Dataset to perform research in software error analysis and reliability modeling. Data collection forms were developed to provide a basis for collecting software productivity and error rate data during software development, conversion and maintenance. A study was performed to determine the data parameters needed to support research activities, model development, and analysis efforts in the areas of software research.

5.2 Database Evaluation

Quantitative metrics to measure and evaluate the quality of existing and future data for modeling and analysis efforts were developed. These metrics will facilitate data collection efforts and modeling activities.

The metrics developed are named integration and coverage. Integration (I) is a numeric measure of the similarity between the datasets in a database. A high degree of similarity ($I \approx 1$) facilitates comparisons of results of a given model among different datasets in a database. Coverage is a graphical metric which measures what model can be driven by the data contained in a database. In this technique, the cells of a two-dimensional array (whose axes are (x) model and (y) data parameter) are filled with the name of the dataset and data element that can supply the data parameter for the model. Matches between data parameters and data elements can then be determined.

These metrics were used to determine the adequacy of the data in the Baseline Software Dataset to do comparisons across a wide variety of projects and to determine if the database contains data elements as required by the various software reliability models and other analytical techniques.

The evaluation determined that the integration factor for the Baseline Software Dataset was $I = 0.28$, a low value that indicates that this database is not well suited to cross comparisons among datasets. Database coverage is high for models that utilize software problem and software modification data, but there are a variety of error analysis and reliability models that cannot fulfill their data requirements from the available data elements.

5.3 Productivity Data Collection Forms

Two data collection forms were designed for software life cycle productivity-related data. The first form, the Project Summary Form,

describes a project's software development environment. The second form the Component Summary Form, describes the software components of a project, resources expended during its development and any noted discrepancies. Component data may be collected at several levels such as system, functional group, and module. In general, the forms have been designed to identify key data collection areas and yet be flexible enough to accommodate additional project-specific data.

The Project Summary Form is used to collect data concerning (1) the general nature of the project, (2) the project priorities and constraints, and (3) the software development technologies used in the project. The data elements characterize the project. Priorities and constraints may be chosen from several common ones listed on the form, or project-specific entries may be noted. A list of software development technologies is also provided. Additional entries may be added.

The Component Summary Form is used to collect data concerning (1) the software engineering characteristics of the component, (2) the resources expended throughout the life cycle of the component and (3) the discrepancies detected in the the component's life cycle. Complexity, type, description, size, programming language, and mode of construction describe the component in software engineering terms. The resources expended on the component are recorded by life cycle phase and time in terms of person hours and computer hours. The number of discrepancies found in the component are recorded by life cycle phase.

The purpose of these forms is to collect project and component level data that is common to most software development projects. When these forms are used as a tool for collecting data for specific quantitative software engineering models or methods, additional project-specific forms should be developed. To develop these special forms, DACS "Software Engineering Research Review Quantitative Software Models," SRR-1, can be used to determine the data parameters to be collected for the specific application.

These forms and their instructions are being distributed by the DACS to invite feedback from the software research and development community.

5.4 Conversion Data Collection Forms

The economics and logistics of software inventory conversion are major procurement issues. In many cases, the cost of the software conversion equals or exceeds the cost of the hardware procured. When a large multi-site hardware upgrade is involved, conversion may take years and require a sizeable dedicated programming staff to complete. Conversion is a relatively new area of interest to software engineers, but one that promises to present a whole new range of problems. At this time, little is known about the types of problems likely to occur during a conversion, the frequency with which they occur across projects, their severity, and how they may be corrected or prevented. By collecting data on past and present conversion efforts, information can be compiled which can be used for identifying problems and their demographics. A database of conversion data will help to conduct feasibility studies, estimate costs for performing

conversions, identify conversion cost drivers, and help to establish cost-benefit relationships for conversion aids and tools. The DACS is seeking to establish such a database which will be made available to the software engineering community.

The DACS has designed three data collection forms to be used during software conversion. The first form, the Software System Overview (SSO) Form seeks to capture those types of data which describe the general nature of the conversion effort. The second form, the Detailed Resource Expenditure (DRE) Form, seeks to capture data which could be used to identify the personnel effort and machine usage required to perform the conversion. The Conversion Problem Report (CPR) Form is used to identify problems encountered, as well as the method of detecting and correcting errors. By filling out a CPR as soon as a problem surfaces during a conversion effort, the chances of any modification being overlooked can be minimized. These forms have also been distributed to the software community to invite user feedback.

5.5 Data Needs for Software Productivity and Reliability

Computer Sciences Corporation performed a study under subcontract to the DACS for the purpose of relating data parameter requirements with various productivity, reliability, and complexity models [SRR1-79]. The results served to facilitate the generation of data collection formats and data requirements lists for use by project managers and software developers involved in the collection of data.

The report [SRR1-79] contains summary descriptions of software productivity, reliablility, and complexity models. The summary descriptions include data required by the models, their major assumptions, and the basic uses of the models. Matrices are included for each model category which correlate the data parameter requirements with the models.

The DACS then performed an extension of this study to determine the data parameters needed to support research activities, model developments, and analysis efforts in the areas of software reliablity, software error analysis, life cycle cost analysis, software productivity, and complexity. The results of this study were summarized in two technical papers written by members of the DACS staff which emphasized the need for the establishment of software experience data collection standards [DUVALL-80], [GLOS-80].

The study resulted in the following conclusions:

(1) Data collection standards must be developed to encompass consistent definition of data and parameters; types of software errors/faults/failures and the degree of their criticality; categorization of faults at the time of their discovery; uniform and comprehensive software problem report formats; operations and maintenance phase data collection (data to be related to execution time); and criteria for discrete time operations versus continuous data streams.

(2) Collection of fault correction data must be expanded to include identification of the likelihood of introducing new errors into the modified program and/or total system; time frame and man power required for correction; and effectiveness of the applied corrections. Collection of fault correction data must be carried into the operations and maintenance phase in a manner consistent with collection during the development phase.

(3) Automated tools should be utilized wherever and whenever possible for data collection. Tools should be designed and/or improved so as to collect data required while intruding as little as possible into the consciousness of development personnel. Such tools would produce more reliable data since they cannot be forgotten, ignored, or incorrectly completed as forms frequently are.

(4) Finally, data collected should be gathered into a central repository such as the DACS to be made available to researchers and practitioners, and hopefully, contribute to the advancement of the state-of-the-art in software technology.

SECTION VI

TASK 5 DEVELOP AN INPUT PROCESSING FUNCTION FOR DATA

6.1 Introduction

The purpose of this task was to review, evaluate, summarize, reformat, and enter newly acquired data into the DACS Software Experience Database. Procedures were established to determine the applicability, accuracy, completeness, validity, and compatibility of the data.

The functions performed in this task may be divided into two catagories; 1) The review and evaluation of data to be acquired or which has already been acquired; and 2) The summarization and reformatting of data for entry into the database.

Figure 6-1 graphically represents the flow of input processing procedures developed for data acquired by the DACS. This figure illustrates that the function is not limited in its scope to the period of time between data aquisition and data analysis. Instead it is continuously providing interaction with both of those programs.

6.2 Review and Evaluation of Data

The review and evaluation of data begins with the identification of data available for acquisition. The evaluation process continues after the data has been acquired, through data analysis activities. The processes involved aid in the determination of the strengths and weakness existing in each dataset.

Procedures and methods have been established to evaluate each of the datasets with respect to the following criteria:

- Applicability to the types of analyses performed at the DACS and by its users and to the types of software development process measures which are being studied.

- Compatibility of the newly acquired or soon to be acquired data with existing data in the DACS database in terms of the various parameters recorded and the methods by which they were recorded.

- Completeness of the dataset or the extent to which the parameters were actually recorded and the amount of data which is not present in the dataset, but which is required for basic analysis.

- Accuracy of the data with respect to how well the parameters, as they were defined, were actually recorded; and the precision of certain data items, i.e., project size recorded as 5000 lines of code vs. being recorded as 5234 lines of code.

- Validity of the methods used to collect and record the data with respect to the intended purposes of the data collection effort and

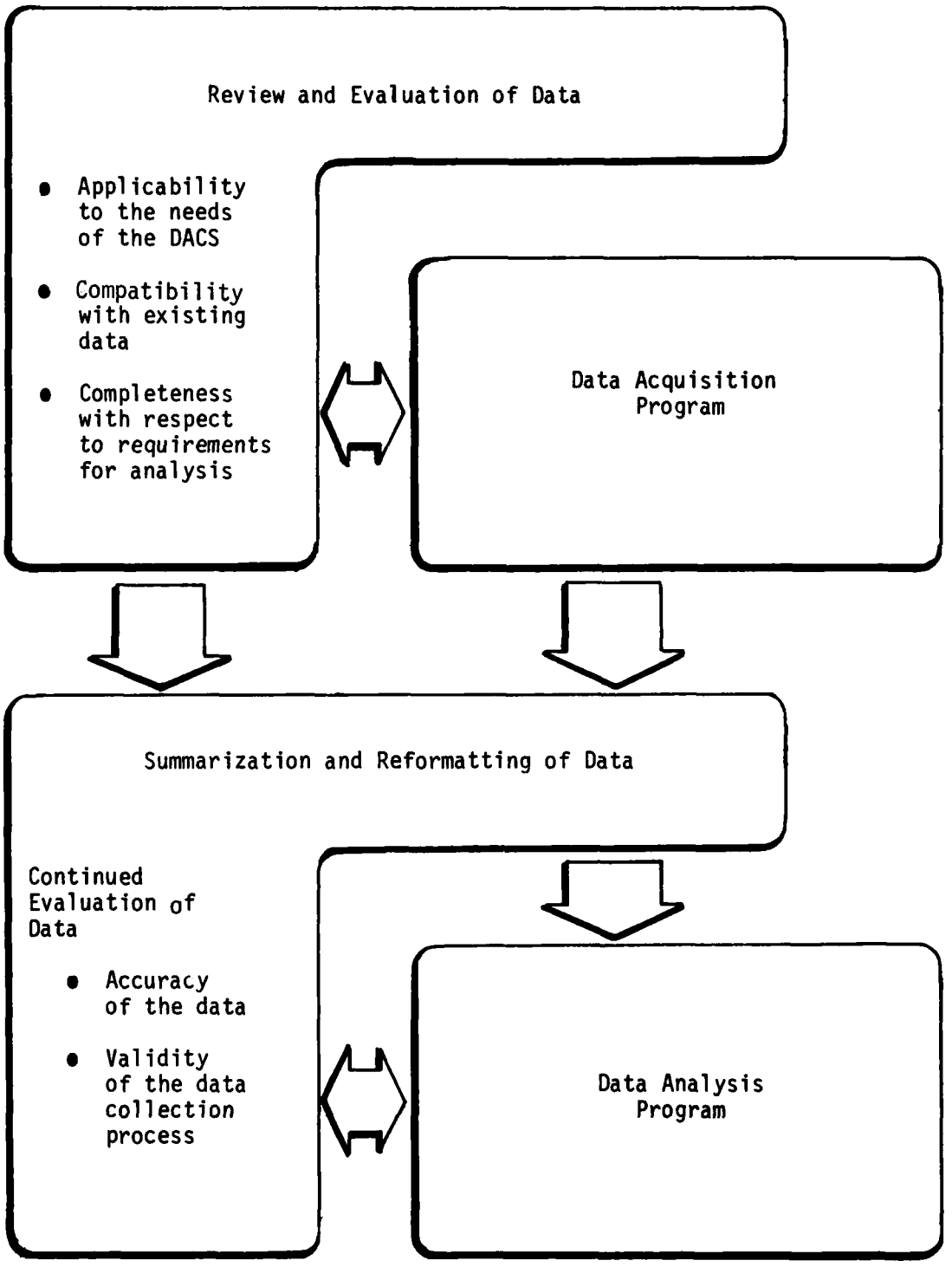


FIGURE 6-1

THE INPUT PROCESSING FUNCTION FOR DATA

the relationships between those purposes and the objectives of analysis activities at the DACS.

These procedures and methods cover a spectrum of possible approaches to evaluation of newly acquired datasets. The applicability of their use to each new dataset is determined empirically and only those procedures and methods necessary for proper evaluation before input into the DACS Software Experience Database are applied.

6.2.1 Determination of Applicability and Compatibility

Prior to acquiring data, some effort is expended to determine if the data is applicable to the needs of the DACS or its users. Often, possible sources of data are located through published research papers or technical reports. In this instance, the published material often provides a great deal of information. This information includes:

- The amount of data collected
- The specific parameters collected
- The methods used in data collection
- The purpose for the data collection activity
- An indication of parameters derived from the raw data
- An indication of computed parameters (as opposed to collected)
- References to studies performed on the data

At this point, an evaluation based on available documentation is made to determine whether or not data that has already been collected should be acquired. This evaluation is based primarily upon the data being applicable to the types of studies being performed in the software engineering field. The parameters collected are compared with the requirements for various models. Data collection methods are examined to determine if collection procedures for the dataset under consideration are consistent with procedures used to collect previously acquired data. For example, specific taxonomies may have been used to classify faults for each dataset; the data may require recasting if the taxonomies vary significantly from each other.

After a dataset has been acquired, individual data items are compared with existing data through data dictionaries or examination of actual data to more closely determine compatibility. As this level compatibility is based upon factors such as the units in which data is recorded, the components of various totals recorded, and the precision with which data is recorded. This evaluation not only determines the dataset compatibility, but also provides input to the summarization and reformatting processes in the data input function.

At this point in time, COVERAGE metrics, described in Section 5.2, may

be computed for the dataset. This metric is used to answer the question,

"Does a database or dataset contain data elements that can supply the data requirements of the various models the software engineer or researcher is using?"

This question is important because its answer determines the scope of the analysis effort that a given dataset can support. Unless a dataset can provide all the data requirements of a model, the model cannot be utilized.

COVERAGE is the ability of a database to supply data for a class or a group of models. COVERAGE concepts are communicated by graphical techniques. A matrix is formed where rows denote the various data parameter requirements for a particular group of models, and columns represent the different models in the group. Table 6-1 represents a COVERAGE matrix for a hypothetical situation involving five models and the sets of six parameters containing all parameters required by any of the models.

TABLE 6-1

SAMPLE COVERAGE DATA MATRIX

	Model 1	Model 2	Model 3	Model 4	Model 5
Parameter 1	1, a * 2, j 3, m	1, a * 2, j 3, m		1, 1 * 2, j 3, m	
Parameter 2		* 2, k	* 2, k		* 2, k
Parameter 3			* 1, c 2, g		* 1, c 2, g
Parameter 4			*		
Parameter 5				1, b * 2, f 3, z	1, b * 2, f 3, z
Parameter 6				* 3, t	

Each cell of the matrix may contain up to three types of symbols. An asterisk in a particular cell indicates that the particular data parameter is required by the particular model. A number corresponds to the particular dataset within the database containing the parameter. A letter following the number represents specifically which data item within the dataset, contains the parameter.

In this example, the parameter required by model 1, is supplied by each of the three datasets 1, 2, and 3. parameter 4, which is required by model 3 is not supplied by any of datasets.

After a dataset has been determined applicable to the needs of the DACS and compatible with data already acquired by the DACS, more detailed analysis is performed.

6.2.2 Determination of Completeness

An important criteria used in the evaluation of a dataset is that of completeness. A study was performed by the DACS to answer the question:

> "Does the database provide for consistency of data across datasets to enable comparisons of analytical results?"

This question is important because the ability to compare results across projects is essential in order to identify factors in the software development cycle that contribute to software quality. Additionally, comparable data from a wide variety of sources is required to further refine, test, and validate present and future analysis and modelling techniques.

The result of this study was a metric, INTEGRATION, also described in Section 5.2, which is a numeric measure of the similarities between the datasets in a database. A high degree of similarity ($I \approx 1$), facilitates comparisons of results of a given model or other analysis activity, among different datasets within a database.

Table 6-2 displays an example of a data matrix which may be used in the computation of an integration metric. The names of twelve datasets are listed across the top of the matrix as projects 1-12. The data items contained in those datasets are listed down the side as data items A-J. Each element of the matrix contains either a one (1) or is empty. A one (1) in a particular element of the matrix indicates that the data item referencing the row was recorded in the dataset for the particular project referencing that column. In the example in Table 6-2, data item B was recorded for project 1, and was not recorded for project 5.

The integration metric is computed from this table in two steps. The first step involves determining the percentage of all project datasets which contain a given data item. This percentage (Integration Factor) is computed for each data item being examined. In this example, data item A occurs in each of the datasets resulting in a integration factor of 1.0 (100%), while data item J occurs in only three of the twelve datasets resulting in an integration factor of .25 (25%).

TABLE 6-2

SAMPLE INTEGRATION DATA MATRIX

Data Item	Project 1	2	3	4	5	6	7	8	9	10	11	12
A	1	1	1	1	1	1	1	1	1	1	1	1
B	1	1	1	1		1	1	1	1	1	1	1
C	1	1	1	1			1	1	1	1	1	1
D	1	1	1		1	1	1	1	1	1		1
E	1	1	1	1		1	1	1	1	1		1
F	1	1		1		1			1	1	1	1
G	1	1		1		1	1	1	1	1		1
H	1	1		1			1	1			1	
I	1	1	1	1	1	1			1	1		1
J	1							1				1

The second step involves averaging all of the integration factors for each of the data items. This resulting average of the individual integration factors is the integration metric for the dataset or database under consideration. In this example, the average of each of the integration factors is 0.611 (61.1%).

The integration factor may also be expressed graphically , as in Figure 6-2. This graph shows the frequency distribution of integration factors over data items. This Figure was obtained from the data presented in Table 6-2. This figure shows that a common data item appeared in 10 out of 12 of the projects, 30% of the time, while a common data item appeared in all of the projects only 10% of the time.

This metric may be used as a measure of the completeness of a dataset in that it is dependent upon the existence of each of the data items in each of the datasets being examined. An ideally "complete" dataset would be one in which each required data item was recorded for each project. This situation would yield an integration factor of one for the entire dataset. Integration factors may be computed for each newly acquired dataset and thus new datasets could be compared against ideally "complete" datasets.

6.2.3 Determination of Accuracy and Validity

The determination of the accuracy and validity of a particular dataset prior to input into the database is primarily based upon the historical information gathered throughout the data acquisition effort and on contact with the personnel responsible for collecting the data and/or any of the research community who have already performed analysis on the data. Accuracy is primarily based upon how well data items, as defined, were recorded. Some of the criteria used include:

- Precision: The number of significant digits to which a parameter was recorded; e.g., the same project may have size recorded as 4589 source lines or as 5000 source lines.

- Redundancy: The ability to cross check data items with other data items within the database; e.g., the total number of changes to a project may be recorded in a summary file or computed from individual change report data, or from historical development data.

- Thoroughness: The property describing the ability of the data collection procedures to perform their specified functions; e.g., the development history of a project may be defined by the data collection process as being recorded weekly and actual data may not indicate this.

- Vulnerability: The susceptability of the data to being recorded inaccurately; e.g., effort-descriptive data items may be recorded as lower than actual to improve the appearance of the development effort.

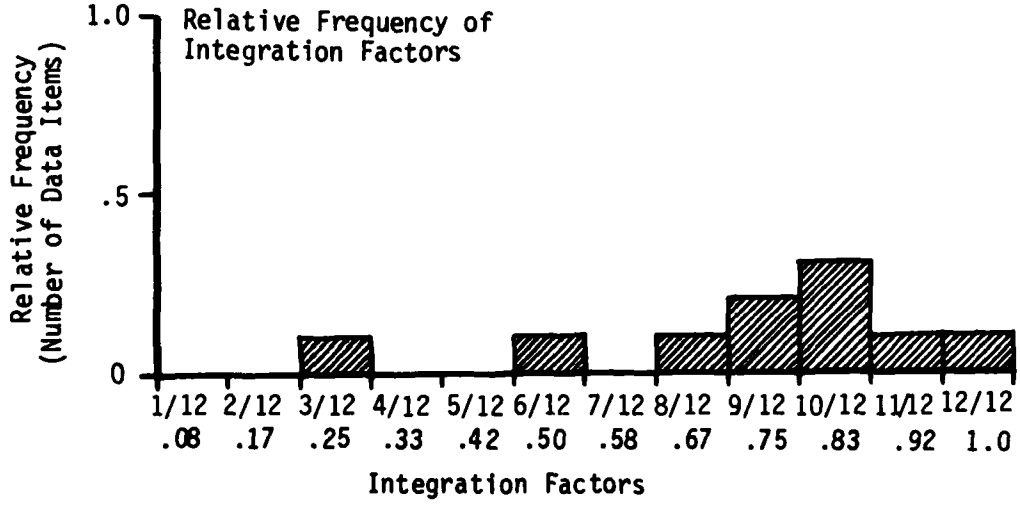


FIGURE 6-2

PROPORTION OF DATASETS CONTAINING A DATA ITEM

The validity of a dataset primarily relates to the methods used in the data collection process. Three basic aspects of the data collection process are examined for validity.

(1) Process: The data collection procedures used may be examined to determine if the parameters collected actually were the parameters desired.

(2) Schedule: The data collection schedule may be examined to determine if data collection activities are frequent enough to accurately reflect the software development activity.

(3) Parameters: The parameters being collected may be examined to determine if they are sufficient to use in the intended analysis, either by the developing organization or the DACS.

The evaluation of the accuracy and validity of a dataset continues after it has been incorporated into the DACS Software Experience Database. Often, the results of analysis activities lead to negative conclusions as to the desirablity of acquiring a particular dataset. The lack of software experience in particular areas, however, often provides the motivation to set asidé judgments made about the accuracy or validity of a dataset. Such datasets will be included in the database and subjected to further analysis until more desirable data becomes available.

In the situation where a source of data is identified before software development has started, the applicability of the data to be collected to research, and the compatibility of the data with existing data, may be maximized through DACS involvement in of the data collection process. As discussed in Section 3.3, the DACS is actively seeking to impact the data collection procedures being used for projects undergoing development.

6.3 Summarization and Reformatting of Data

Once a dataset has been reviewed and evaluated, it may be rejected or entered in the Software Experience Database. Data acquired is rarely in the form most suitable for the Software Experience Database. Datasets vary in the record and file format of the data, the units in which parameters were recorded, the level of detail or summarization used to record the data, or the various taxonomies used to classify certain parameters. The summarization and reformatting methodologies developed by the DACS seek to preserve the integrity of the data while making it suitable for analysis purposes.

The Standardized Software Productivity Dataset, introduced in Section III, provides an example of such required summarization and reformatting.

Each of the newly acquired datasets requires varying amounts of summarization or reformatting prior to entry into the database. The methodologies used to accomplish this are discussed below.

6.3.1 Summarization Methodologies

Summarization of parameters within a dataset occurs when the parameter in question is recorded in greater detail than what is needed. This situation occurs when a parameter is recorded in increments over a period of time, and what is required is a total number of occurrences for a time frame. The situation also occurs when a taxonomy used to classify a parameter is more specific than that required or presently in use. Summarization is the process whereby one new parameter is computed from several existing parameters or a new classification scheme is derived from a more detailed taxonomy.

In summarizing occurrences, the relative time period involved is important. The phases of the software life cycle provide the reference to time in this situation. Since development phases are readily agreed upon, these time frames may generally be associated with newly acquired datasets. The summarized metric will then be associated with a particular phase or phases of the software life cycle, such as the total number of changes made to a project during integration, system testing, or validation and verification testing. The summary metric would then be computed by summing the occurrences during the particular phases, ignoring occurrences outside of the specified time period.

6.3.2 Reformatting Methodologies

Reformatting of parameters within a dataset occurs when the parameter in question is recorded in different units than what is needed or currently in use. This situation occurs quite often in the case of project size and project effort, either of which may be recorded in any number of different units. Reformatting would then be the process whereby data measured in one unit is converted to other units.

In reformatting data, the definitions of the units involved and historical information specific to the dataset under consideration are both important to the reformatting process. A clear understanding of the definitions of the units involved is required to select proper conversion equations. For instance, if one were to convert source code measurements to executable machine code measurements, one would want to be concerned with whether or not the source code measurements included comments. Sources of historical information pertinent to the dataset include the conversion equations used by the software development organization or data collection organization, and published studies of similar data where similar reformatting conversions were required. Reformatting would thus involve examining the selected unit definitions and selecting or computing the proper conversion equation based on an understanding of the definitions.

Mention should be made of the physical processing of the data. In order for the data to be used, it must be read from tape or input by hand and stored on disk on the RADC HIS 6180. Personnel at the DACS have established a working knowledge of the utility and MDQS subsystems which automate considerably the tasks involved in physically processing the data.

The utility subsystems are used to read and copy tapes containing

data. The subsystem allows for a wide range of tape formats to be read.

The MDQS subsystem is used to reformat records once a tape has been read and copied onto disk. Directory and data definition files are created for the acquired and existing data files and an MDQS routine is developed which equates the common parameters and computes totals, averages, and other statistics as required.

6.4 Processed Datasets

Several acquired datasets have been processed through portions of the input process function. Many others are in various stages of being acquired and are being examined and reviewed.

The datasets which have been acquired and accepted for further analysis are: The NASA/SEL Dataset; the RADC Productivity Dataset; the Software Reliability Dataset; the BSDS Dataset; and the V&V Dataset. The NASA/SEL Dataset has been accepted primarily because of the accuracy and detail to which portions of the development process were recorded, although integration and coverage metrics have not been computed. The RADC Productivity Dataset was accepted because its size and compatibility with acquired and future datasets. The BSDS Dataset is being maintained for future addition to the database. The V&V Dataset has been recently acquired and added to the database. The error taxonomy used for the V&V Dataset will provide enhancements to the error data already in the database.

6.4.1 Standardized Productivity Dataset

The Standardized Software Productivity Dataset represents one result of the input processing functions. The parameters selected for inclusion into the Standardized Software Productivity Dataset, as defined in Table 3-1 were selected based upon the availability of the parameters in the datasets acquired by the DACS. Each of the datasets has undergone various portions of the input processing function.

Two basic types of parameters are recorded in this dataset. These are: parameters which measure the development process and parameters which record the development environments.

Development process parameters include such items as size, effort, project duration, and number of trouble reports. These parameters are usually recorded in acquired datasets in a usable form. The size parameter for NASA/SEL project data was reformatted (recomputed) to eliminate unmodified reused code from this figure.

Development environment parameters represent the usage of various software engineering tools and techniques during the development of a software project. These include the extent to which MPPs were used. A wide variety of methods have been used to record parameters of this type, and therefore, these parameters usually require reformatting. These parameters were reformatted from available data and background information for the NASA/SEL Dataset. Development environment parameters represent the

constraints which special environmental factors placed on the software development. At this point in time, only the primary language used on the project is recorded in the standardized database.

6.5 Summary

The preceeding sections describe input processing functions developed for data identified and/or acquired by the DACS. It should be noted that these functions are influenced by the type and quality of data available in the software engineering community today. As uniform collection methods become the norm and units of measure become more standardized, or parameters to be collected evolve into a standard set, these input processing functions will be modified or augmented.

SECTION VII

TASK 6 CURRENT AWARENESS PROGRAM

7.1 Introduction

Concurrent with the establishment of the DACS, a comprehensive and vigorous current awareness program was established. This program has been maintained throughout the three-year pilot operation of the DACS.

The program has had two purposes; (1) to keep the DACS user community informed of the latest and most significant developments in software technology and software engineering, and (2) to inform its user community of new products and services offered by the DACS. The DACS has utilized many means to implement the current awareness program. These include:

- Publication of the DACS Newsletter
- Publication of the DACS Bulletin
- Presentations at Conferences and Symposia
- Establishment of Contacts throughout the software engineering community through active participation in professional organizations
- Placement of press releases and announcements concerning DACS products and professional activities in professional journals, newspapers, and magazines circulated to the software engineering community
- Publication and dissemination of informational materials designed and developed by the DACS staff

Each of these activities will be discussed in the following sections.

7.2 DACS Newsletter

The first step in implementing the current awareness program was the publication of the DACS Newsletter. The Newsletter is employed as the primary means for the dissemination of current information to the DACS user community. It contains the following: synopses and critiques of significant, newly acquired reports/articles, summaries of new R&D programs, listings of future conferences/symposia, summaries of significant technological break-throughs and significant new technological applications, and highlights of other developments within the Center's fields of interest. It is distributed free-of-charge to both government and non-government personnel having an interest in the disciplines served by DACS.

Since its first publication in October 1978, six monthly newsletters

and nine quarterly newsletters have been published.

The DACS Newsletter is now sent to the 2896 individuals on the mailing list and is also distributed at conferences attended by DACS personnel. Figure 7-1 illustrates the DACS Newsletter whose name, format, and general content was conceived and developed by DACS staff members.

7.3 DACS Bulletin

The first DACS Bulletin was first published in April 1979, and since then 17 issues have been published. Figure 7-2 illustrates a typical issue of the DACS Bulletin.

The Bulletin has been regularly distributed on a limited basis to RADC/CO personnel. Regular distribution is approximately 50 copies. The DACS Bulletins are usually a treatment in greater depth than space allows in the Newsletter. Certain issues of particular interest to our general user community have been publicized in the Newsletter and distributed to requesting individuals although not identified as a Bulletin.

7.4 Information Publications

Literature on the DACS products and services has been prepared and distributed to users and potential users of the center. A general information packet is sent in response to an initial inquiry on the DACS. This packet contains a copy of the current Newsletter, a copy of the paper entitled "An Introduction to the Data and Analysis Center for Software," individual flyers which contain order forms and describe the various products, and the guide entitled "Bibliographic Services and Custom Searches." This guide, BIB-1, was designed to acquaint the users of the DACS with procedures required to obtain a custom search of the DACS bibliographic database. As of August 14, 1981, 521 information packets had been distributed in response to requests for information on the DACS. As of the same date, over eleven hundred copies of BIB-1 had been distributed.

7.5 Technical Symposia Activities

Table 7-1 describes twenty-two technical presentations which were made at software engineering related technical symposia. Eight papers were published in workshop or conference proceedings. The major subject areas included in these papers are:

- An Introduction to and Status Report on the DACS
- A Summary of the DACS Report Entitled "Quantitative Software Models"
- The State-of-the-Art in Software Reliability
- A Review of the DACS Database Evaluation Methodology
- Software Engineering Technology Transfer

DACS NEWSLETTER

Data & Analysis Center for Software

RADC/ISISI
Griffiss AFB, NY 13441
315/330-3937
Autovon 587-3396

Volume III, Number 1
March 1981

EDITOR'S NOTES

The DACS Newsletter for 1981 will be contained in Volume III. This is the first newsletter of that volume. The volume number will change consecutively with each year. The colors for each season that the newsletter is distributed are: Spring — green, Summer — yellow, Fall — driftwood tan, and Winter — blue.

This issue contains a Predictive Software Cost Model Study Update; a review of the recently published book, Techniques of Program and System Maintenance; an update to the DACS Thesaurus; a description of NASA/SEL Data Set; a summary of the DACS evaluation of the Doty Models; and an announcement of the 1981 Minnowbrook Workshop.

PREDICTIVE SOFTWARE COST MODEL STUDY UPDATE

In the September 1980 issue of the DACS Newsletter we reported a study funded by the Air Force Systems Command, Avionics Laboratory of Air Force, Wright Aeronautical Laboratories. The purpose of the study was to determine the feasibility of developing a model to predict avionics embedded software support costs and to generate a roadmap for development of such a model.

The study indicated that actual model development was indeed feasible. A follow-on contract was awarded to System Consultants, Inc. (SYSCON) on September 1, 1980. SYSCON is currently in the process of collecting support cost data from several Air Force Logistics Centers. The collected data will be analyzed and used to develop and validate the model. Model development and validation should be complete in late 1982.

The AFWAL project engineer is Daniel Ferens, AFWAL/AAAS-2, Wright Patterson AFB, OH 45433; Autovon 785-5346 or (513) 255-5346.

Copies of this study can be obtained from NTIS, 5285 Port Royal Road, Springfield, VA 22161 (703-487-4650). The report number is AFWAL-TR-80-1-56, Volumes I and II. The NTIS order numbers are ADA 088-476 and ADA 088-477.

TECHNIQUES OF PROGRAM AND SYSTEM MAINTENANCE

Girish Parikh of Ethnotech, Inc., has edited a collection of papers on software maintenance and published them in a book entitled, *Techniques of Program and System Maintenance* (1980). The book is available from Ethnotech, Inc., P.O. Box 6627, Lincoln, NE 68506 (402/489-8861), price is $25.

The publication of the text was motivated by the growing awareness that software maintenance costs comprise approximately 70% of overall software life cycle costs. Recent studies predict that they will increase at the rate of approximately 22% per year through 1985.

The re-occurring theme of the text is that the software maintenance process should be reorganized into a hierarchical structure comprised of distinct functional activities which utilize standard software maintenance methodologies, techniques and tools patterned after modern software development methodologies. Parikh believes that the utilization of proven modern development methodologies, techniques and tools will improve the quality and reduce the expense of the maintenance process.

The book is divided into seven sections, five of which are collections of papers. The sixth and seventh sections contain an annotated bibliography and an index, respectively. The topics covered in the first five sections include the following: 1) "Maintenance: The Problem and Perspective," 2) "Practical Tips for Maintainers," 3) "Management Considerations and Techniques," 4) "Structured Technologies and Maintenance" and 5) "The Future of Maintenance."

The collection of papers in Section I defines the terminology, problems and the scope of software maintenance technology. The writers in this section agree on one thing: a formalized maintenance technology is required to reduce cost overruns. Several also mention that the problem of a negative label being applied to the maintenance function often causes maintenance programmers to feel they are stuck in "second-rate" jobs.

Parikh has included one of his own papers which reiterates his hierarchical view of software engineering which he presented in an earlier article.

The second section, "Practical Tips for Maintainers," contains examples of the methodologies which Parikh advocates in the first section. The papers are written by professionals who share their own experiences in maintaining software. Several interesting tools and techniques are discussed.

In the third section, papers by several managers cover techniques which have improved their maintenance operations. One writer reviews the improvements made at his company when they reorganized the maintenance functions into a completely separate activity from other activities in the programming department. They structured the work of their maintenance staff into three categories: repairs, revisions and enhancements. The writer maintains that the separation of duties had a positive effect on morale within the department.

Section IV covers structured programming techniques and how they effect maintenance. The writers are not in agreement as to whether structured technologies will result in time or cost

The Data and Analysis Center for Software is Operated by IIT Research Institute for the Rome Air Development Center

FIGURE 7-1

NAME, FORMAT AND CONTENT OF THE NEWSLETTER WERE DEVELOPED BY DACS PERSONNEL

DACS BULLETIN

Volume II, Number 3
April 1981

Data & Analysis Center for Software

RADC/ISISI
Griffiss AFB, NY 13441

315/336-0937
Autovon: 587-3395

NEW GAO REPORT AVAILABLE

Software Maintenance

The US General Accounting Office (GAO) reviewed the status of computer software maintenance and summarized their findings and recommendations in a report entitled "Federal Agencies' Maintenance of Computer Programs: Expensive and Undermanaged." The document is available for review at the DACS or may be ordered free-of-charge from:

US General Accounting Office
Document Handling and Information Services Facility
P.O. Box 6015
Gaithersburg, MD 20760
Telephone (202) 275-6241
Order Number: AFMD-81-25
Date: 26 February 1981

A summary of the salient points reported follow:

- The director of the General Services Administration's Software Development Office has estimated that the Government spends at least $1.3 billion annually on software maintenance (excluding embedded weapons system software).
- Estimates of Government spending on software range as high as $6 billion a year.
- The current cumulative Federal investment in software probably exceeds $25 billion.
- Half of a programmers time is devoted to maintenance.

The Data and Analysis Center for Software is Operated by IIT Research Institute for the Rome Air Development Center

FIGURE 7-2

THE DACS BULLETIN FOCUSES ON A SINGLE TOPIC OR REPORT

- Some estimates say that up to 60% of all ADP dollars will be spent on software maintenance in the future if the present growth rate continues.

- One recent DoD study showed that the cost of development for Air Force avionics software averaged about $75 per instruction while the cost of maintenance corrections of deployed software has ranged up to $4,000 per instruction.

- Government managers were not managing software maintenance as a function. The absence of maintenance management is due, at least partly, to the absence of definitions and guidelines.

- The absence of maintenance management is also shown by the common lack of data on costs and types of maintenance being done and the common absence of standards or goals for software maintenance performance.

- Increased management attention to the following problem areas could reduce costs.

  (1) Excessive user-requested modifications.
  (2) Inadequate definition of requirements in system development phase.
  (3) Inadequate or missing documentation.
  (4) Inadequate control of contractor software developments.
  (5) Limited use of software tools and techniques in the development and maintenance of software.

Recommendation Summary

- Costs associated with the software maintenance function should be monitored and recorded.

- Software maintenance should be identified and managed as a discrete function.

- Definitions of software maintenance should be standardized.

- Guidance specifically and explicitly directed at techniques for reducing Federal software maintenance should be developed and published.

This report also provides a Provisional Checklist for Software Maintenance Management which summarizes ways to control and reduce software maintenance costs.

This report is highly recommended to individuals/organizations that are involved in software acquisition, development, and maintenance.

FIGURE 7-2 Cont'd

TABLE 7-1

DACS TECHNICAL PRESENTATIONS AND PAPERS

DATE	AUTHOR(S)	TITLE	PRESENTATION
August, 1981	Christopher Turner Lorraine Duvall	Software Engineering Data Validation Synthesis, and Analysis at the DACS	*Fourth Minnowbrook Workshop on* Software Performance Evaluation Syracuse University's Minnowbrook Conference Center Blue Mountain Lake, NY August 11-14, 1981
	Lorraine Duvall	Data & Analysis Center for Software: A Positive Approach to Technology Transfer	Fifth Annual Summit Meeting of the Nation's 12 Leading Soft-Experts Washington, D.C. August 6-7, 1981
October, 1980	Shirley Gloss-Soler, Lorraine Duvall	Data Needs for Software Engineering	*Thirty-sixth Annual National* Electronics Conference Chicago, IL October 27-29, 1980
August, 1980	Shirley Gloss-Soler	The DACS Data Acquisition Program	Third Annual Minnowbrook Conference on Software Performance Evaluation Blue Mountain Lake, NY August 19-22, 1980
June, 1980	Lorraine Duvall	Software Engineering Technology Transfer	DoD Embedded Computer Resourses Conference Atlanta, GA June 3-5, 1980
May, 1980	Jon Martens	The Role of an Information Analysis Center in Software Engineering Technology Transfer	National Computer Conference Anaheim, CA May 19-23, 1980
January, 1980	Lorraine Duvall, Jon Martens	Data Needs for Software Reliability Modelling	Proceedings 1980 Reliability and Maintainability Symposium San Francisco, CA January 22-24, 1980
December, 1979	Lorraine Duvall	Technology Dissemination	AIAA Computer Conference Washington, DC December 3-5, 1979
November, 1979	Jon Martens, Lorraine Duvall	Designing a Software Data Compendium	Fourth Software Engineering Workshop, NASA Goddard Space Flight Center Greenbelt, MD November 19, 1979
	Shirley Gloss-Soler	Data & Analysis Center for Software	Association for Computing Machinery Rochester, NY November 14, 1979

TABLE 7-1 Cont'd

October, 1979	Lorraine Duvall	Software Reliability Program Concepts	National Electronics Conference Chicago, IL October 29, 1979
	Jon Martens	Metrics for Evaluating Quantitative Software Modelling Databases	Workshop on Quantitative Software Models for Reliability, Complexity, and Cost Kiamesha Lake, NY October 9-11, 1979
	Lorraine Duvall	A Review of the Evaluative Literature on Software Reliability Modelling	
	Lorraine Duvall, Shirley Gloss-Soler	A Summary of the Second Minnowbrook Workshop on Software Performance Evaluation	
August, 1979	Shirley Gloss-Soler	Data & Analysis Center for Software	Sixteenth Annual Computer Personnel Research Conference Princeton, NJ August 16-17, 1979
July, 1979	Lorraine Duvall, Shirley Gloss-Soler	Status of Data & Analysis Center for Software	Second Minnowbrook Workshop on Software Performance Evaluation, Syracuse University's Minnowbrook Conference Center Blue Mountain Lake, NY July 31 - August 3, 1979
June, 1979	Shirley Gloss-Soler, Lorraine Duvall	An Introduction to the DACS Software Engineering Bibliographic Database	Eighteenth Annual Technical Symposium Information Systems---Effectiveness for the User Gaithersburg, MD June 21, 1979
March, 1979	Lorraine Duvall	A Software Reliability Review	Proceedings Second National Reliability Conference Birmingham, England March 28-30, 1979
January, 1979	Lorraine Duvall	Data & Analysis Center for Software	Proceedings of 1979 Annual USAF Academy Computer Related Information Systems Symposium (CRISYS) Colorado Springs, CO January 24-26, 1979
December, 1978	Lorraine Duvall	Introduction to the Data & Analysis Center for Software	ACM 1978, Washington, DC December 6, 1979
September, 1978	Lorraine Duvall	An Introduction to the Data & Analysis Center for Software	First Minnowbrook Workshop on Software Performance Evaluation, Syracuse University's Minnowbrook Conference Center Blue Mountain Lake, NY September 20, 1978
	Lorraine Duvall	An Introduction to the Data & Analysis Center for Software	Third Software Engineering Workshop, NASA Goddard Space Flight Center Greenbelt, MD September 18, 1979

- Data Requirements for Software Engineering

Technical presentations proved to be both an effective way to inform the technical community of the concerns and activities of the DACS and to attract new users. They were also an effective way to identify new areas of concern in the community the DACS serves and to identify sources of data for the data acquisition program, as well as inputs for the Newsletters and Bulletins.

Interaction with colleagues at the technical symposia often includes discussion of areas not tracked or closely followed by the DACS. However, the knowledge of who is working in a particular area may often be of as much value to DACS users as if the DACS had references and materials on such topics.

7.6 User Awareness

User awareness is an important aspect of center management in that it provides managers and technologists with clear and available descriptions of the products and services. During this reporting period we have disseminated this information through the Newsletters, informationa' brochures, and participation in conferences and workshops (as discussed in previous sections).

Throughout the reporting period DACS personnel have consistently maintained a high quality Newsletter. We have found that the Newsletter is an effective tool in communicating to users the availability of the various products and services, both existing and planned.

Throughout the reporting period, DACS personnel have also maintained a high visibility at conferences and workshops. These activities are extremely effective both in announcing and explaining new products and services, and in obtaining user feedback on existing products and services. The nature of conferences and workshops allows for the type of interaction most helpful in enhancing products and services.

7.7 Summary

As a result of the vigorous conduct of the current awareness program, several benefits have accrued to the DACS.

- DACS personnel have established an elaborate network of contacts throughout the software engineering community and are regularly invited to make presentations at conferences and symposia which will further advance the current awareness program.

- DACS personnel have developed an in-depth knowledge of the needs and concerns of the DACS user community. This knowledge is used to prepare and publish Newsletters and Bulletins which are both informative and useful and used as input to decisions on new product development.

- Through participation in technical symposia and workshops, DACS personnel have acquired a broad base of information regarding the activities of individuals and organizations in the field of software engineering, as well as a reputation for the DACS as a clearinghouse for such information.

SECTION VIII

TASK 7 THE SCIENTIFIC AND TECHNICAL INFORMATION BASE

8.1 Introduction

Scientific and Technical Information (STINFO) consists of documented information concerning the state-of-the-art and technology aspects of the computer software field. STINFO usually includes technical reports, trade journal publications, proceedings of conferences and symposia, theses, texts, product descriptions and specifications. The DACS also includes as STINFO descriptions of ongoing software technology research for which reports may not yet have been produced. These two types of STINFO serve as input to two information databases maintained by DACS, the Software Engineering Document Library and the Software Engineering Research Projects (SERP) Database.

In order to identify and support the terminology used in these databases a glossary and thesaurus have been developed.

The status of this information base is discussed in this section of the report.

8.2 The Software Engineering Document Library

The DACS Library has been established to provide a readily accessible source of comprehensive information on the state-of-the-art in software engineering as well as a means of channeling that information to those people in the software engineering community who can make use of it in their day-to-day activities of developing, maintaining, and managing software. The bibliographic collection is composed of texts, technical reports, thesis, journal articles, proceedings and other documents relating to software engineering, reliability, cost and quality factors, maintainability, and other topics deemed appropriate.

The DACS Software Engineering Bibliographic (SEB) Database contains the document descriptions and abstracts of documents in the DACS library. The database is made available to members of the software engineering community chiefly in the form of computer-generated subject bibliographies. The datafiles are implemented to provide rapid retrieval of individual document citations. The datafiles are accessed through the MDQS implemented on the Honeywell 6180 computer. This database management system allows searching on any of the items in a bibliographic citation and/or assigned keywords and outputs bibliographies of documents retrieved during the searching process. Searching may be done on-line or in batch mode.

As of 14 August 1981, the DACS had approximately 2800 documents relating to software engineering; 2200 of these documents have been indexed, their bibliographic citations coded, keypunched and placed online. In addition, 1999 of these documents have been described in two hard-copy bibliographies that contain author, subject and keyword-in-context (KWIC) indices in addition to complete citations and abstracts for each document.

8.2.1 Identification and Acquisition of Documents

8.2.1.1 Identification of Documents

A coordinated program has been established to identify and acquire published STINFO. The major sources of documents and the methods used to identify pertinent documents for acquisition are shown in Table 8-1.

The decision to order/acquire a particular document is usually based upon an evaluation of a description of the document rather than upon evaluation of the document itself. When descriptions of seemingly pertinent documents have been identified, the descriptions are evaluated by two engineering personnel for suitability for the DACS. The document is then acquired through the acquisition procedures outlined in Section 8.2.1.2 When the ordered document arrives a final evaluation as to its technical excellence and relevance to the scope and charter of the DACS must be made.

Table 8-2 lists the types of documents in the online database, the number of documents of each type, and the percent of each type. Note that 85% of the documents are journal articles and papers.

8.2.1.2 Acquisition of Documents

Once a document is deemed suitable for acquisition, the document description is tested for uniqueness of author and title against the current author index and current author-in-process index. The document description is then reviewed to determine the agency from which it is available.

If a document is available from a document distribution center or a publisher (i.e. NTIS, DTIC, IEEE Service Center, ACM Order Dept., etc.) the document is assigned an order number, the author and title is recorded on the DACSLOG data sheet for online entry at the end of the day, the description is entered on a document description form, and the order form is mailed to the distributing agency.

TABLE 8-1

MEANS OF IDENTIFYING SOFTWARE ENGINEERING DOCUMENTS

SOURCE OF DOCUMENT	TYPE OF DOCUMENT	IDENTIFICATION METHOD
U.S. Government Nat'l Technical Information Service (NTIS)	Unclassified gov't reports prepared by gov't agencies and their contractors or grantees	Regular review of publicizing instruments NTIS Technical Notes
Defense Technical Information Center (DTIC)	Classified or limited distribution reports on R & D done by DoD or its contractors	DTIC Technical Abstracts Bulletin
General Accounting Office	Investigative reports on activities of government agencies	DACS is on distribution for reports pertinent to software technology
Naval Publications and Forms Center	DoD standards, regulations and specifications	Index of Specifications and Standards
RADC	Technical Reports	Distribution list
Professional Society	Papers, conference proceedings, journals, tutorials	Membership in Society; Attendance at selected conferences, workshops, symposia; Review catalogs of publications
Publisher of Periodicals directly relevant to software engineering	Journal articles, papers	Subscription
Publisher of Texts	Texts	Catalog or announcement or book review in professional journal
Colleges and Universities	Theses and Dissertation	Periodic review of Dissertation Abstracts published by University Microfilms
Information Retrieval Services	Papers, conference proceedings, journals, tutorials	Subscription; periodic review of Engineering Index and Computers & Control Abstracts
Authors	Papers, technical reports	Catalog; announcement; book review in professional journal; attendance at selected conferences, workshops, symposia; user inquiries

TABLE 8-2

DOCUMENT TYPE

TYPE OF DOCUMENT	NUMBER OF DOCUMENTS	PERCENT OF COLLECTION
BIBLIOGRAPHY	0013	00.6
JOURNAL ARTICLE	0508	23.6
TECHNICAL REPORT	0339	15.7
TEXT	0012	00.6
PAPER	1250	58.0
STANDARD	0009	00.4
REGULATION	0004	00.2
SPECIFICATION	0003	00.1
MONOGRAPHS	0018	00.8
TOTAL DOCUMENT CITATIONS ONLINE	2156	

If the document availability is unknown, the DACS contacts the author or agency which produced the report in order to acquire information about its availability or to acquire the document directly from the author or agency.

8.2.1.3 Periodicals

DACS subscribes to indexing/abstracting periodicals from government and professional agencies. The document descriptions in these periodicals are reviewed as outlined in Section 8.2.1.1 and relevant documents are acquired by following the procedures outlined in Section 8.2.1.2. Other indexes and abstracts not on DACS subscription lists are reviewed periodically at other IITRI centers or at local academic libraries.

In addition, DACS has initiated subscriptions to journals which regularly publish articles and papers directly relevant to the concerns of the DACS. However, not all articles or papers published in these journals are relevant to the DACS, and therefore, not all articles are processed for inclusion in the database.

A full listing of the various journals and proceedings from which the DACS has obtained articles and papers is contained in Table 8-3

8.2.1.4 Proprietary Information Handling Procedures

The DACS has acquired certain documents which contain proprietary information. All such documents are stamped to indicate that they contain such information upon arrival at the DACS.

In the case of those documents for which the producer wishes to have their existence publicized but their distribution controlled by the producer, the document is processed for the library but filed separately. Use of the document at the DACS is limited to DACS personnel. When the citation appears in a custom bibliography, a statement that the document contains proprietary information and must be requested from the producing organization also appears. In the case of those documents for which the producer does not wish their existence publicized, a separate file is maintained; the document is not processed for the library, and use or discussion of the document is limited to DACS personnel.

8.2.2 Document Processing and Database Update

The DACS has established procedures which insure that all incoming published STINFO is reviewed for technical excellence and relevance to the concerns of the DACS before processing begins. The DACS Thesaurus is used as a guide to determine the relevance to the DACS holdings. Those not meeting the evaluation criteria are returned, discarded or filed without processing.

The major activities are log-in, abstracting, coding, indexing, verification, keypunching, and loading. Figure 8-1 gives a schematic view

TABLE 8-3

BREAKDOWN STATISTICS FOR JOURNAL ARTICLES AND PAPERS

AS OF 08/14/81

NAME OF JOURNAL OR CONFERENCE	NO. PAPERS OR ARTICLES
16TH ANN. COMPUTER PERSONNEL RESEARCH CONFERENCE	0007
17TH ANNUAL TECHNICAL SYMPOSIUM,NBS/ACM	0018
1972 FALL JOINT COMPUTER CONFERENCE	0001
1973 IEEE SYMPOSIUM ON COMPUTER SOFTWARE RELIABILITY	0025
1975 ANNUAL RELIABILITY & MAINTAINABILITY SYMPOSIUM	0005
1975 CANADIAN SRE RELIABILITY SYMPOSIUM	0001
1976 CONFERENCE ON INFORMATION SCIENCES & SYSTEMS	0001
1977 ANN. SOFTWARE LIFE CYCLE MANAGEMENT WKSHOP	0035
1977 ANNUAL RELIABILITY & MAINTAINABILITY SYMPOSIUM	0003
1979 ANNUAL RELIABILITY AND MAINTAINABILITY SYMPOSIUM	0004
1980 ANNUAL RELIABILITY AND MAINTAINABILITY SYMPOSIUM	0009
1ST NATL. CONFERENCE ON SOFTWARE ENGINEERING (1975)	0014
2ND INT'L CONFERENCE ON SOFTWARE ENGINEERING	0084
2ND INT'L CONF. ON SOFTWARE ENGINEERING, ADDENDUM	0018
2ND NATL RELIABILITY CONF.,1979 (BIRMINGHAM, ENGLAND)	0003
2ND SOFTWARE LIFE CYCLE MANAGEMENT CONFERENCE	0001
2ND SUMMER SOFTWARE ENGINEERING WORKSHOP	0004
39TH ANN. CONF. OF THE SOCIETY OF ALLIED WEIGHT ENG.	0001
3RD DIGITAL AVIONICS CONFERENCE	0001
3RD INT'L CONFERENCE ON SOFTWARE ENGINEERING	0047
3RD SUMMER SOFTWARE ENGINEERING WORKSHOP	0004
4TH INT'L CONFERENCE ON SOFTWARE ENGINEERING	0050
4TH SUMMER SOFTWARE ENGINEERING WORKSHOP	0005
ACM 75, PROCEEDINGS OF THE 1975 ANNUAL CONFERENCE	0037
ACM 78,PROCEEDINGS OF THE 1978 ANNUAL CONFERENCE	0034
ACM COMPUTING SURVEYS	0008
ACM INSTALLATION MANAGEMENT REVIEW	0001
ACM/NBS 18TH ANN. TECHNICAL SYMP.,INFORMATION SYSTEMS	0013
AFIPS NAT'L COMPUTER CONFERENCE,1979	0018
AGARD CONFERENCE PROCEEDINGS NO. 261	0008
AIAA/NASA/IEEE/ACM COMPUTERS IN AEROSPACE COF. 1979	0038
AIAA/NASA/IEEE/ACM COMPUTERS IN AEROSPACE CONF. 1977	0058
ANN. COMPUTER RELATED INFORMATION SYSTEMS SYMP.,1977	0006
ANN. COMPUTER RELATED INFORMATION SYSTEMS SYMP.,1978	0005
ANN. COMPUTER RELATED INFORMATION SYSTEMS SYMP.,1979	0008
BELL SYSTEM TECHNICAL JOURNAL	0001
BYTE	0001
COMMUNICATIONS OF THE ACM	0017
COMPCON '75	0004
COMPCON '76	0003
COMPCON '78	0002
COMPUTER	0098

TABLE 8-3 Cont'd

COMPUTER BUSINESS NEWS	0003
COMPUTER JOURNAL,THE	0001
COMPUTER SYSTEMS NEWS	0001
COMPUTERS AND PEOPLE	0002
COMPUTERWORLD	0038
COMPUTING NEWSLETTER	0001
CONFERENCE ON LANGUAGE DESIGN FOR RELIABLE SOFTWARE	0014
CONF. ON SPECIFICATIONS OF RELIABLE SOFTWARE,1979	0019
CONF. ON STATISTICAL COMPUTER PERFORMANCE EVALUATION	0018
DATA PROCESSING	0002
DATABASE	0002
DATAMATION	0024
DEFENSE MANAGEMENT SYSTEM REVIEW	0001
DELTAK TRAINING BULLETIN	0001
EASCON '74	0002
EDP ANALYZER	0012
ELECTRONIC DESIGN	0012
ELECTRONIC NEWS	0001
ELECTRONICS	0006
EVALUATION ENGINEERING	0001
FIRST CANADIAN SRE RELIABILITY SYMPOSIUM, 1974	0001
HARVARD BUSINESS REVIEW	0002
IBM SYSTEMS JOURNAL	0011
IEEE SPECTRUM	0001
IEEE TRANSACTIONS ON COMPUTERS	0006
IEEE TRANSACTIONS ON RELIABILITY	0017
IEEE TRANSACTIONS ON SOFTWARE ENGINEERING	0229
INFOR - CANAD.JRNL OF OPERATIONAL RES.& INFOR.PROCESNG	0001
INFORMATION PROCESSING 74	0003
INFORMATION PROCESSING LETTERS	0001
INFORMATION SYSTEMS NEWS	0001
JOURNAL OF SYSTEMS AND SOFTWARE	0007
JOURNAL OF SYSTEMS MANAGEMENT	0006
MICRO & MINI SYSTEMS, SYMP. ON TRENDS & APPL.,1976	0012
MICROPROCESSORS & MICROCOMPTERS, 2ND ED.	0002
MILITARY ELECTRONICS/COUNTERMEASURES	0002
MINI-MICRO SYSTEMS/MAY 1980	0001
MRI SYMPOSIUM ON COMPUTER SOFTWARE ENGINEERING 1976	0038
NAECON '75 RECORD (NATL AVIONICS & ELECTRICAL CONF)	0005
NATIONAL COMPUTER CONFERENCE, 1974	0003
NATIONAL COMPUTER CONFERENCE, 1975	0001
PAPERS ON PROGRAM TESTING	0013
PRACTICAL STRATEGIES FOR DEVELOP'G LG SOFTWARE SYSTEMS	0008
PRICE/PARAMETRICS CONF./ISPA	0011
PROCEEDINGS OF THE IEEE	0001
PROCEEDINGS OF THE NATL. ELECTRONICS CONFERENCE	0010
PROCEEDINGS OF THE SECOND ARMY SOFTWARE SYMPOSIUM	0028
PROCEEDINGS,1970 ANN. SYMPOSIUM ON RELIABILITY	0001

TABLE 8-3 Cont'd

PROCEEDINGS,1975 INT'L CONFERENCE ON RELIABLE SOFTWARE	0059
PROCEEDINGS, 1969 ANN. SYMPOSIUM ON RELIABILITY	0001
PROCEEDINGS, COMPSAC 77	0050
PROCEEDINGS, COMPSAC 78	0101
PROCEEDINGS, COMPSAC 80	0004
PROCEEDINGS,IFIP WC CONSTRUCTING QUALITY SOFTWARE	0029
PROCEED. OF THE SOFTWARE & QUALITY ASSURANCE WORKSHOP	0029
PROCEED., INTERSERVICE/INDUSTRY TRAINING EQUIP. CONF.	0001
RADC/ARPA CONF. ON SOFTWARE VERIFICATION & VALIDATION	0018
RESEARCH DIRECTIONS IN SOFTWARE TECHNOLOGY	0021
SIGNAL - JOURNAL OF AFCEA	0001
SOFTWARE ENGINEERING NOTES (ACM SIGSOFT)	0033
SOFTWARE - PRACTICE AND EXPERIENCE	0103
SYMPOS. ON HIGH COST OF SOFTWARE,MONTEREY, CA	0009
TECHNIQUES OF PROGRAM & SYSTEM MAINTENANCE	0030
TOOLS FOR EMBEDDED COMPUTING SYSTEMS SOFTWARE	0030
TUTORIAL: AUTOMATED TOOLS FOR SOFTWARE ENGINEERING	0025
TUTORIAL: SOFTWARE DESIGN STRATEGIES	0001
TUTORIAL: SOFTWARE MANAGEMENT	0031
UNPUBLISHED PAPER OR REPORT	0005
WORKSHOP ON QUANTITATIVE SOFTWARE MODELS	0029
WORKSHOP ON SOFTWARE TESTING & TEST DOCUMENTATION	0022

NOTE: THE TOTAL DOCUMENTS IN THIS TABLE WILL BE GREATER THAN THE SUM OF THE PAPERS AND JOURNAL ARTICLES IN THE PRECEEDING TABLE. THE CAUSE IS THAT SEVERAL PAPERS HAVE APPEARED IN MORE THAN ONE PUBLICATION; TO PROVIDE ALTERNATE SOURCES FOR OUR USERS, WE HAVE INCLUDED REFERENCES TO ALL SOURCES OF WHICH WE ARE AWARE.

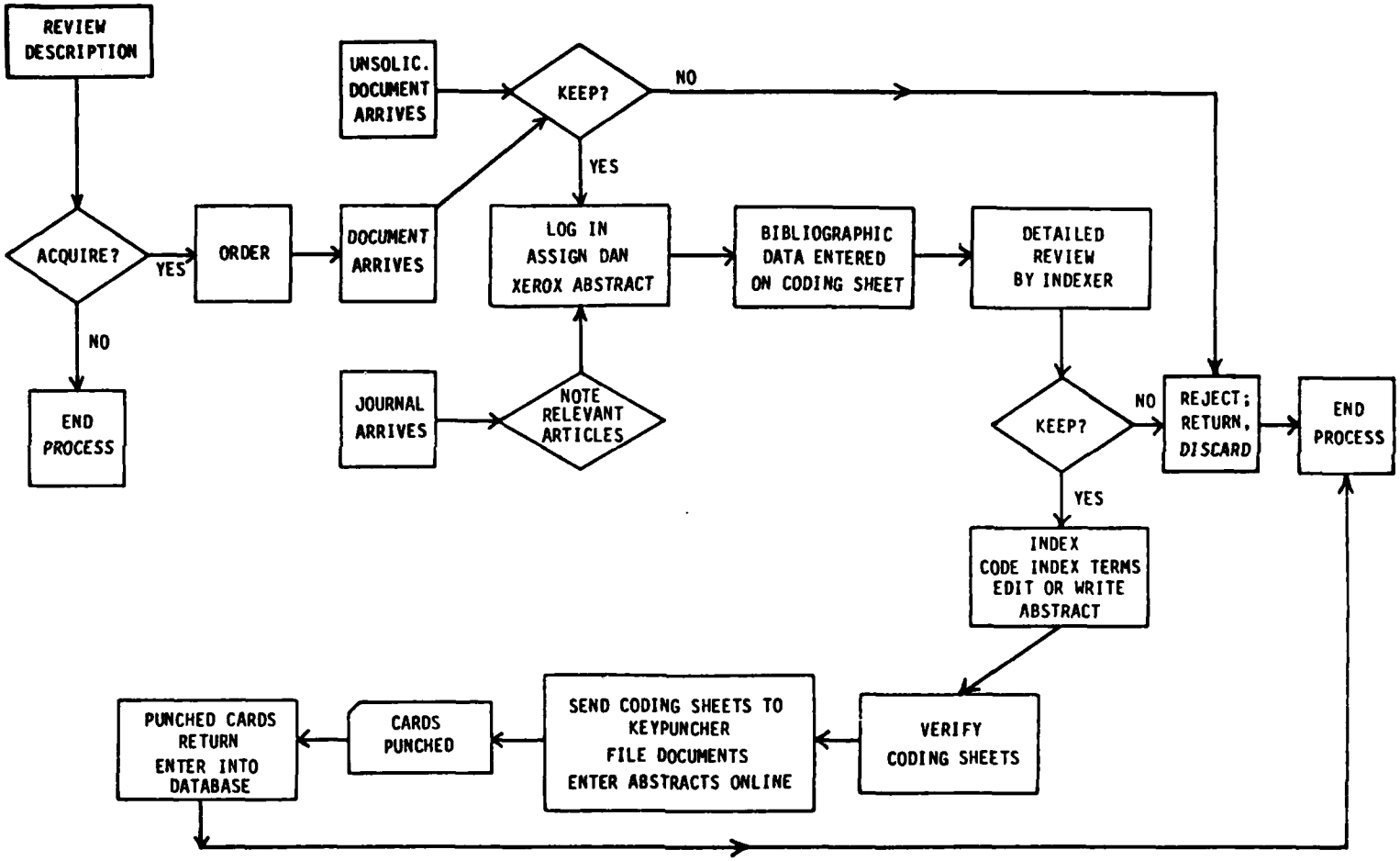


FIGURE 8-1

DOCUMENT PROCESSING PROCEDURES

of the processing procedures.

8.2.3 Retrieval of Bibliographic Information

The DACS functions as a centralized repository and provides custom searches upon receipt of a request from a user. The request may be made during a visit to the center, by a telephone call, or by a written request. When the search has been completed, the resulting bibliography is given or mailed to the user who made the request.

It should be noted that the retrieval language of MDQS is procedurally oriented and syntactically demanding for all but the simplest retrieves or where any degree of formatting of output is needed. To alleviate the problems inherent in such a system, DACS personnel have written a series of parameterized queries. These queries are maintained online and were designed to be utilized by non-programming personnel. The person running a search need only log-on, type the letters "MDQ " and the name of the file containing the query, and then type "RUN". The MDQS system will request three parameters, (the topic of the search, the search strategy, and the retrieval qualifiers), actuate the retrieval and direct the resulting bibliography to an off-line printer. Clerical personnel at the DACS have been trained to actually run the queries which produce the custom searches after the search parameters have been supplied by either one of the DACS engineering staff or by the requestor of the bibliography.

8.3 SERP Database

The DACS maintains the SERP Database to provide a computer-accessible source of information about recent or ongoing research in the field of software engineering.

The initial work in defining the database and collecting information on the first group of projects was done under subcontract to the DACS by Computer Sciences Corporation. IITRI has expanded and upgraded the database which is being maintained on a continuing basis to provide a clearinghouse for research being done in software technology.

Projects in the database are those involving software technology research, such as the development or evaluation of programing languages, models or software tools, and research related to software engineering methodologies such as modern programming practices. These projects should not be confused with software development projects which involve the production or acquisition of software programs or systems. DACS also maintains a database of data from software development projects and acquires data through the DACS Data Acquisition Program (see Section 3.3).

The information provided in this database enables researching managers and developers to fulfill many needs. Researchers are able to access information about current software engineering research projects in order to exchange information, to find who is doing research in areas of similar interests, and to become more familiar with the state-of-the-art in software technology and software engineering research. The software

acquisition managers and developers are able to plan for future utilization of newly emerging software tools and methodologies. Research sponsoring organizations are able to identify qualified sources for software research efforts and eliminate duplicate research efforts.

Information about completed projects is extracted from published reports describing the research at the time such reports are processed for the document library.

However, there is often a long time period between the initiation of a research project in software engineering and the publication of the report summarizing the research and its findings. Knowledge of the existence of such projects can avoid duplication of effort and allow preliminary or partial results to be applied to other research efforts. The DACS has developed mechanisms to identify such projects in advance of report publication. These mechanisms will be discussed in Section 8.3.1.

As of 14 August 1981, the DACS has acquired project descriptions from approximately 200 projects relating to software engineering; 175 of these descriptions have been indexed, coded, keypunched, and placed online.

8.3.1 Identification of Unpublished Research Project Information

Most of the projects in the research projects database will produce reports which will be included in the DACS document library and some will produce data, either embedded in the report, or as a deliverable in itself. It is desirable to obtain the data and the documents at the time they are summarized or published, thus saving valuable time rather than waiting for the documents to be processed through another document center before they can be identified by the DACS.

Projects which are candidates for inclusion in the database are identified chiefly by using the following major information sources:

Commerce Business Daily (CBD)

Electronic News (Contract Awards)

DTIC (Work Unit Summary)

ACM Software Engineering Notes

Computing Newsletter

DACS Newsletter

Personal Contacts

8.3.1.1 Commerce Business Daily

The CBD is desirable as it makes possible identification prior to contract awards so that reports can be received early in the life cycle.

The first action is to establish contact and determine the extent of applicability of software technology information that may emanate from the program. This is accomplished by sending form letters and questionnaires to contracting agencies. The purpose of these questionnaires is to obtain information on the status, type of software technology research to be done, and the nature of reporting that is expected to be produced during the contract.

8.3.1.2 Electronic News

Electronic News contains the "Contract Awards" column where final government contract awards are identified. The awards listed in the column are compared to projects originally identified in the Commerce Business Daily and the proper organizations are contacted for information describing the research projects.

8.3.1.3 DTIC Work Unit Summary

Another source of information which is used to identify software R&D projects is the DTIC Work Unit summary. This summary identifies programs which are new or ongoing. A search was requested from DTIC and projects which were software engineering research oriented were selected and entered into the SERP Database.

8.3.1.4 Software Engineering Notes

ACM Software Engineering Notes is the publication of the ACM Special Interest Group on Software Engineering (SIGSOFT) and regularly contains abstracts of research being conducted in software technology. Information on projects of interest to the DACS is entered into the database in much the same manner as for that obtained from DTIC.

8.3.1.5 Computing Newsletter

An article was prepared and sent to the editor of the Computing Newsletter. The article described the Software Engineering Research Projects Database and requested that software engineering researchers publicize their research projects in the SERP Database.

8.3.1.6 DACS Newsletter

The DACS used its quarterly Newsletter to publicize the need for more research project information to be added to the SERP Database.

8.3.1.7 Personal Contacts

Personal contacts made through work on professional committees and attendance at conferences, workshops, and symposia as well as interaction with DACS users has proven to be an excellent source of information about software engineering research projects.

8.3.2 Special Effort to Identify Non-Government Research Projects

A special effort was made to obtain project information on non-government (i.e. industrial and academic organizations) in-house projects. The information sources cited in Section 8.3.1 were used in order to contact those organizations that would be able to contribute this information. The effort was begun in April 1981 and was concluded in July 1981.

As a result, we received a total of six solicited responses and one unsolicited response. Of these three were projects funded through non-government agencies, and the remaining projects were funded through government contracts or grants.

It should be noted that most companies contacted were reluctant to volunteer the required information. However, DACS will continue to publicize the need for an information exchange within the software engineering community. An early study designed to evaluate the usefulness of obtaining information from a research project database, found that users felt the information obtained particularly helpful in terms of keeping up with other work in their field, finding bibliographic leads, avoiding duplication in the preparation of grant applications and being able to contact other scientists directly for exchange of information [HERSEY-63].

Similar responses were received as a result of a survey conducted by the U.S. General Accounting Office in 1973 [GAO-73].

8.3.3 Processing Information for the SERP Database

Information submitted on the form shown in Figure 8-2 is evaluated for relevance to the concerns of the DACS, for consistency and for completeness. Missing information is re-requested from the principal investigator(s) or sponsoring agency as appropriate. When the information gathering is complete and it is determined that the project is relevant, the project description is logged, abstracted, coded, indexed, keypunched and entered into the SERP Database in a manner similar to that used for the bibliographic database (see Section 8.2.2).

8.4 The DACS Glossary

There were two objectives in compiling the DACS Glossary. The first objective was to record the terminology currently in use. The second was to provide users of the DACS software engineering bibliographic services a means of ensuring that they are using terms from the DACS Thesaurus for their retrievals in the same way as the terms were used for indexing. A closely related objective was to promote consistency in indexing new documents for the collection.

The first version of the DACS Glossary was published by the DACS in October 1979. Most of the terms and definitions were compiled from the software engineering literature. This DACS Glossary contained 1123 terms, 1100 of which had one or more definitions and 23 of which are cross-referenced to other terms. This first version of the DACS Glossary was the most popular of the early DACS products with 1498 copies being

SOFTWARE ENGINEERING RESEARCH PROJECT DESCRIPTION

PROJECT NAME:

PRINCIPAL INVESTIGATORS:

PERFORMING COMPANY (NAME, PHONE NUMBER, ADDRESS):

PROCURING ORGANIZATION:

CONTRACT/GRANT NUMBER:

START DATE:

COMPLETION DATE:

CONTRACT/GRANT VALUE:

PROJECT ABSTRACT:

PROJECT DELIVERABLES: (NAME, REPORT NO., TYPE OF REPORT, ORDER NO., PAGINATION, AVAILABLE FROM):

FIGURE 8-2

INFORMATION IS COLLECTED FOR ENTRY INTO SERP DATABASE

distributed between October 1979 and August 1981.

8.4.1 Coordination with the IEEE Terminology Standard

During the period after publication of the DACS Glossary in October 1979, the database has been maintained online. There were two reasons for this. First, the original DACS Glossary consisted of terms taken directly from the literature without being edited for consistency of format and with multiple definitions of different terms taken from different sources. Second, the DACS Glossary was to be coordinated with the software engineering terminology standard being developed by the IEEE Computer Society. The need to refine the definitions in the first version and to also coordinate with the IEEE work on software engineering terminology made it necessary for a DACS staff member to become a member of the task group developing the IEEE terminology standard. The task group had disbanded due to its chairman taking an IEEE position of higher responsibility. Thus, in order to coordinate the DACS Glossary with the IEEE Software Engineering Terminology (SET), it was necessary for a DACS staff member to assume the chair of the terminology task group (TTG) and to reassemble it. Shirley Gloss-Soler, of the DACS, took on that task.

8.4.1.1 Establishing a Working Group

The intent in soliciting a task group was to obtain a balanced representation that would portray the full range of work being done in software engineering as well as a full range of organizational affiliations. Thus, in addition to contacting the seven members of the original working group to see if they wished to renew their participation, a call for volunteers was published in several professional journals. Approximately 150 individuals responded to the call for volunteers. To cope with the communications problems inherent in a group of 150 people, a superstructure, the Terminology Task Group Steering Committee, was formed to coordinate the activities of the 150 volunteers. An effort was made to secure a steering committee membership representing a full range of organizational affiliations as well as a full range of professional expertise. The steering committee is composed of 12 people; Table 8-4 lists both current members and past members together with their organizational affiliations.

8.4.2 IEEE SET and DACS Glossary Interaction

The IEEE SET was scoped to include terms and definitions specific to software engineering and an effort was made to exclude terms whose use was primarily computer science or which was a term borrowed from some other discipline and used no differently by software engineers than by other types of engineers. The DACS Glossary must reflect a broader point of view than the IEEE SET because it must contain terms (and definitions) specific to DoD, terms used to index and retrieve documents for the bibliographic database and reflect the point of view of the Air Force. All work done for and with the IEEE Terminology Task Group has been incorporated into the online database maintained at the DACS which is used as input for both the DACS Glossary and IEEE SET. Thus, the latest version of either is available

TABLE 8-4

STEERING COMMITTEE OF THE TERMINOLOGY TASK GROUP

Shirley Gloss-Soler, Task Group and Steering Committee Chairperson, IIT Research Institute and Data & Analysis Center for Software

Russell J. Abbott, California State University, Northridge and the Aerospace Corporation

Joan P. Bateman, Boeing Computer Services Company

Stephen R. Beason, Digital Equipment Corporation

Milton Boyd, Jr., Digital Equipment Corporation

Kurt F. Fischer, Computer Sciences Corporation

Marilyn S. Fujii, Logicon, Incorporated

Lt. Glenn C. Hughes, II , United States Army

John Ives, Air Force Weapons Laboratory

John McKissick, Jr., General Electric Company

Albrecht J. Neumann, National Bureau of Standards

John Postak, Doty Associates, Inc.

Jane W. Radatz, Logicon, Incorporated

Alan N. Sukert, Rome Air Development Center and General Electric Company

Donald A. Woodmancy, NCR Corporation

on demand.

As a result of interactions with the members of the Terminology Task Group, many terms and authoritative definitions not contained in the SET were obtained for the DACS Glossary. Version 2 of the DACS Glossary was produced in February 1981 and will be published and distributed by the DACS in the fall of 1981.

In addition, the result of a DACS staff member chairing the task group was that the DACS automatically obtained information about standards being developed by approximately 30 other standards-making bodies such as the International Electrotechnical Commission (IEC), American National Standards Institute (ANSI), Electronic Industries Association (EIA), etc. In each case, the DACS was offered an opportunity for one of its staff members to join the working group responsible for drafting the standard, comment upon drafts, maintain liason, or in some other way influence the direction of the evolving standard.

8.5 The DACS Thesaurus

In order to index both the document collection and the project information, the DACS Software Engineering Thesaurus was constructed during the first phase of the contract and is used for both indexing and retrieval purposes. This thesaurus is periodically updated in order to keep it current with the changing terminology in software engineering. The thesaurus is composed of 519 keywords or terms arranged into 49 groups of closely related keywords headed by a cluster term. A listing of the cluster terms is contained in Table 8-5.

Appendix C contains a listing of all 519 keywords in the DACS Thesaurus together with the number of documents to which each keyword has been assigned. Appendix D contains a listing of all 114 keywords from the DACS Thesaurus together with the number of projects to which those keywords have been assigned. There were 12,767 keyword assignments made to 2158 online document descriptions, yielding an average of 6 keywords per document. There were 524 keyword assignments made to 175 online project descriptions, yielding an average of 3 keywords per project.

TABLE 8-5

CLUSTER TERMS

APPLICATIONS
ARCHITECTURE
ARTIFICIAL INTELLIGENCE
CERTIFICATION
COMPLEXITY
CONVERSIONS
COST
CORRECTNESS PROOFS
DATA
DEBUGGING
DOCUMENTATION
DATABASE MANAGEMENT SYSTEMS
DISTRIBUTED PROCESSING
DESIGN
DEVELOPMENT
EDUCATION
ERRORS
FAILURES
FAULTS
FUNCTIONS
HIERARCHY
IMPLEMENTATION
LANGUAGES
MICRO COMPUTERS
MODELS
MANAGEMENT
MINICOMPUTERS
MODULES
MODELLING AND SIMULATION TOOLS
MAINTENANCE
OPERATING SYSTEMS
PERFORMANCE
PROGRAMS
PROGRAMMING
PROCEDURES
QUALITY
RELIABILITY
REQUIREMENTS
SECURITY
SOFTWARE ENGINEERING
SPECIFICATIONS
SOFTWARE TOOLS
STANDARDS
SOFTWARE
TRI-SERVICE
TESTING
VERIFICATION
VALIDATION
VIRTUAL MACHINES

SECTION IX

TASK 8 PRODUCTS AND SERVICES PREPARATION AND DISTRIBUTION

9.1 Introduction

This section summarizes the results of the tasks to produce and distribute the DACS products and services. A characterization of these products and services is presented, along with summarized quantitative information on requests processed. More detailed data on the number and type of requests processed is included in Section 10.2, Improving Center Effectiveness. The products distributed as a result of these requests are summarized in Table 9.1

9.2 Data Subsets

The purpose of this task is to distribute subsets of the DACS database (as discussed in Section 3.2 of the report) to aid in research efforts that require productivity, cost, complexity, error, and change data. These datasets are being used to validate and refine software reliability and software productivity models and methods. The availability of the RADC Productivity Database was announced in the June 1979 DACS Newsletter. As of 14 August 1981, 463 copies of this dataset had been distributed in hard copy report format or on magnetic tape.

To facilitate distribution of this dataset, descriptive literature on the dataset is provided to the user together with instructions for ordering it. The literature contains a check-off list for specifying the sorted order of a computer listing and magnetic tape requirements. This data subset is distributed along with a data dictionary describing the data elements and a copy of the draft report entitled "Software Data Collection and Analysis" which presents a preliminary analysis of the dataset.

The availability of the Software Reliability Dataset was announced in the December 1979 DACS Newsletter. A comparison report entitled "Software Reliability Data" was produced by John Musa. This report contains descriptions of the data and the characteristics of the software systems, listings of the data, and ordering information for obtaining a copy of the data on magnetic tape. As of 14 August 1981, requests for approximately 323 copies of this database had been processed.

The availability of the NASA/SEL Dataset was not announced in the Newsletter, as resources have not allowed us to fully study the options available for subsetting and disseminating the great volume of data present in this dataset. However, as of 14 August 1981, three inquiries for the entire dataset or portions of it have been processed.

The BSDS Dataset and V&V Dataset are not currently being distributed.

9.3 Data Compendiums

TABLE 9-1

DACS PRODUCTS AND SERVICES PRODUCED AND DISTRIBUTED THROUGH 8/1/81

MONTH INITIAL DISTRIBUTION BEGAN	PRODUCT OR SERVICE	TOTAL NUMBER DISTRIBUTED TO DATE
October 1978	Custom Bibliographies and Searches Consulting DACS Newsletter Nelson Report on RADC Productivity Database	706 113 2735* 339
December 1978	DACS Information Packet	513
April 1979	BIB-1 (The User's Guide to Custom Searches) DACS Bulletin DACS Thesaurus Quantitative Software Models (SRR-1)	1162 40* 1162 921
July 1979	RADC Productivity Database-Magnetic Tape or Hardcopy Format Gilb Report	124 88
December 1979	Productivity Data Collection Forms The DACS Glossary	348 1498
February 1980	Software Reliability Database - Hardcopy Report	313
April 1980	NASA/SEL Database Software Reliability Database - Magnetic Tape Format	3 10
February 1981	Conversion Data Collection Forms Monograph on Data Collection with References	16 15
Products Produced But NOT DISTRIBUTED by the DACS	A Review of Software Maintenance Technology (SRR-2) Software Engineering Research Projects Report The DACS Annotated Bibliography DACS Technical Monograph - A Comparison of RADC and NASA/SEL Software Development Data The NASA/SEL Data Compendium Report on Applying the Doty Model to the NASA/SEL data	

* Total Number of names on mailing/distribution list, not total number of copies distributed.

Two software engineering data compendiums were produced during this reporting period. The purpose of these compendiums was to profile the software engineering data that has been collected by the NASA Goddard Software Engineering Laboratory during software development projects at that facility.

Each software development project is described in terms of its purpose, size, life cycle dates and development methodologies, tools and techniques. Data from computer run analysis forms are used to profile the purposes and results of the projects' computer runs. From change report forms, the data elements such as types of errors and effort for error changes are profiled. Several resource expenditure forms are used to produce project profiles regarding number of computer runs, computer time and manpower hours by task.

Several other features are included in the compendium to enable potential users of the data to determine the relevance of the data to their applications. A data dictionary and an evaluation of the completeness of the data by project and data type, are included among the features.

The first data compendium was produced in June 1980 to cover data collected and recorded at the NASA/SEL through approximately January 1980. The second data compendium was produced in April 1981, to enhance methods by which data was presented, and to include data collected through November 1980.

9.4 Technical Reports

Several state-of-the-art reports were produced during this reporting period. The first report, entitled "Quantitative Software Models," (SRR-1), was compiled from a Data Parameters Report produced by Computer Sciences Corporation under subcontract to the DACS. SRR-1 is a compendium of attributes of software cost/productivity, reliability/error analysis, and complexity models. The availability of this state-of-the-art report was announced in the March 1979 DACS Newsletter. As of 14 August 1981, 921 copies of this report had been distributed to the software technology community.

The second report is entitled "Software Maintenance Technology." This study was performed by Computer Sciences Corporation under subsontract to the DACS which focused on the state-of-the-art in software maintenance technology as it is described and defined in the open literature and technical reports [CSC-79]. . The purpose in conducting the technology review was to develop a comprehensive statement about the maintenance techniques and tools in use today and to describe how they support the activities associated with computer software maintenance. The computer program maintenance environment was also examined and is described through discussion of topics related to implementation of software maintenance technology. It is hoped that the results of this study will foster further research into software maintenance principles and ultimately lead to definition of a software engineering discipline encompassing all aspects of computer program development and maintenance.

Material was selected based on its relevance to the subject of software maintenance and the date it was published. Generally, only papers and articles published since 1974 were selected. For reports and books, a publication date of 1976 or later was observed. There were exceptions for material that was deemed to be particularly relevant or unique in content.

One early realization during the study was that the terminology of software maintenance technology was not consistent, possibly due to the rapid development and changes that have occurred in the industry. As a result, a complete glossary of terms is included in Appendix A of the study report.

The report is a practical guide on software tools in terms of categorization, potential use and benefit, and experiences in using the specific tool. A set of maintenance technology functions is defined including configuration control, operations monitoring/evaluation, redesign, code production and analysis, verification and validation, testing and integration, and documentation. Forty maintenance tools and techniques are then defined for each technology function, and an introduction as to how they relate to Operations and Maintenance (O&M) activities is provided.

A principal result of this study is the observation that there has been very little determination of the effectiveness of the tools and techniques on the O&M phase of the software life cycle. This maintenance technology report was published as a RADC technical report and distributed through NTIS. No data is available on the number of these reports which have been distributed.

Several other reports were generated as a result of analysis activities performed during this reporting period. Each of these reports is individually discussed in Section IV. As of 14 August 1981 none of these reports have been formally distributed to users.

9.5 Newsletter

The DACS Newsletter is meant to service our users by providing current information regarding state-of-the-art in Software Technology, operations of the center, the the availability of products and services offered by the DACS, and the addition of products or services. The DACS Newsletter has been distributed to users since the inception of the center. The Newsletter was distributed on a monthly basis until December 1979 when distribution was changed to quarterly. As of 1 August 1981, 2735 users were on the Newsletter mailing list.

9.6 Informational Database Products

Several products were developed by the DACS from the STINFO (Section VIII) databases maintained at the DACS. These databases are:

- SET Database

- SEB Database

- SERP Database

The DACS Glossary is a listing of the entries in the (SET) Database. The Glossary provides the user with definitions of various terms and phrases related to the field of software engineering. Two editions of the Glossary have been compiled during the reporting period, the most recent edition (March 1981) containing over 1000 entries. As of 14 August 1981, 1498 requests for the Glossary have been processed.

The DACS Annotated Bibliography is a hard-copy version of the information contained in the SEB Database. This document contains an annotated list of software engineering literature and several indices to that literature. During the reporting period, we have produced one annotated bibliography and one addendum to that bibliography. Both documents together contain nearly 2000 entries. The bibliographies have not been distributed outside of DACS. The DACS offers, as a service, custom searches on the SEB Database on any topic desired by a user. Two documents, "Custom Searches" (BIB-1) and the "Software Engineering Thesaurus" provide the user with quick access to this service. As of 14 August 1981, 1162 user requests for either a custom search or the Thesaurus have been processed.

The "Directory of Software Engineering Research Projects" was produced in July 1981. The document contains a listing of the entries in the SERP Database which includes descriptions of research projects designed to further the state-of-the-art in software engineering, an index of principle research personnel, a keyword index and a thesaurus. The information in this document is complementary to the document citations in the SEB Database mentioned above. This document has not been distributed as of 14 August 1981.

9.7 Technical Inquiries

Technical inquiries to the DACS are received and processed on a daily basis. These inquiries are received in many forms: by letter, telephone call, visit, or by use of the bibliographic request form contained in BIB-1. The information requests have ranged from the very specific to general questions on software engineering methodologies. Table 9-2 contains the description of a representative sample of the inquiries we have received. The total number of technical inquiries received since inception of the center (from 1 December 1978 through 14 August 1981) was 661. This number does not reflect requests for specific DACS documents or requests for placement on the Newsletter mailing list.

A technical inquiry may be answered in the following one or more ways:

- A custom bibliographic search on the subject area of interest is performed.

- A preliminary analysis of the subject literature is made and summary information prepared.

- A subset of the DACS database is produced.
- Relevant DACS literature is distributed.
- Referrals to other sources are provided.

The DACS has provided 706 custom bibliographies to users. The information needs which prompted these searches, have ranged from broad information requests whose result could be described as a file dump to very narrow requests such as "decision table programming in FORTRAN."

The search profile below indicates the 15 most popular requests. Forty-six additional search topics were requested 15 or fewer times and included such topics as design, data collection, modern programming practices, productivity, embedded computers, and software tools.

Search Topic	Number of Requests
Costs	57
Quality Assurance	33
Testing	32
Quality Factors/Metrics	27
Reliability	27
Software Tools	25
Verification and Validation	24
Maintenance	22
Conversions	20
Development	20
Management	19
Design	17
Errors and Faults	16
Modelling and Simulation Tools	16
Language	16

9.8 Miscellaneous Services

In addition to the previously mentioned DACS products and services, we have produced and distributed a summary of the "Second Minnowbrook Workshop on Software Performance Modeling," edited and distributed copies of a paper by Tom Gilb entitled "Controlling Maintainability - A Quantitative Approach for Software," distributed bibliographies of RADC/CO Software Technology reports, and took an active role in organizing sessions and workshops related to software technology.

TABLE 9-2

SAMPLE DACS TECHNICAL INQUIRIES

Researching certification of computer programs, equipment, and personnel

User needs specific information on the processing and validation of meterorological (weather) data

We are establishing an internal verification and validation group and we need data on software errors, specifically software errors by instruction type

Require information on code verification, i.e. tools that can be used to certify that minimum testing standards have been met

Need information on manpower resource estimation

Would like to know about any techniques to evaluate technical proposals concerning software development, both on a "stand-alone" basis and a hardware basis (i.e., complete system)

This office would appreciate being placed on your distribution list and would like further information regarding the Data Analysis Center for Software

Please send information describing the DACS charter and accomplishments

User needs information on life cycle costing figures or projections and maintenance costing data on the benefits of distributed processing systems

Need information on assembly language vs HOL programmer productivity

Custom bibliography on the verification and validation of computer software

Information needed to prepare a user's acceptance criteria list for a system under development and acceptance testing

Interested in software life cycle cost models

Documentation to accompany testing - formats of test plans/specifications at various stages

We would appreciate your sending copies of the various forms used for collection of software data for analysis

Am looking for any information on product safety - the impact on safety caused by software problems

TABLE 9-2 cont'd

Would like data on frequency of errors for real-time systems
Would like list of references on software quality, reliability, and testing
Need to develop standards and procedures for testing (unit, acceptance, and system)
Need information on estimating manpower resources required for software projects
Reviews weapon systems status prior to full scale production - needs background information on error management, reliability, and quality
Would like information on estimating the size of C^3I systems, functional groups, and modules
Doing a study on life cycle cost models for an Air Force human resources lab; would like information on life cycle cost models
Would like list of references to be used in formulating and implementing software reliability and quality assurance programs
Am a government contractor - doing a large programming job. Need to set up internal standards for design, coding, programming and also evaluation of completed programs
Would like information on the use and benefits of specific programming tools
Need information on software fault detection as well as fault tolerant operations and recovery; I am specifically interested in techniques used to structure software in order to accomplish the above
Need to know the cost impact of implementing a MIL-S-52779 QA plan, especially interested in cost impact during the development phase of the life cycle
Would like a list of references on design methodologies
We are specifically interested in samples of standards and/or procedures for collecting error (failure) information to be used in procuring digital flight control software; we do not want it for reliability estimation but to determine what kinds of tools and techniques need to be developed to prevent/detect these errors

TABLE 9-2 Cont'd

Am looking for survey articles on software acquisition and cost estimation
We have an urgent need for definitions of software engineering terms
We are presently performing a tool survey and would appreciate your assistance with regard to a bibliographic search of your database in conjunction with our efforts
My department has, as its principal responsibility, the quality assurance control and monitoring function on a large SWS Navigation Subsystem; the inclusion of computer software to this quality discipline on this subsystem has been a recent addition, directed by the Navy; any suggestions that would enhance the effectiveness of our quality assurance efforts would be appreciated
Need pratical information on sizing and timing estimation methods - current information requested
One of UNIDO's (United Nations Industrial Development Organization) problems is funding the development of duplicate software packages because minimum consideration is given to portability, maintainability and reliability; please send me information on these aspects of software
Involved in a large USAF language conversion effort - Can't find anything in the literature
The determination of factors which contribute to software quality and the application of techniques to measure these factors
My present research is concerned with development of tools for predicting trouble spots in software development projects; what data do you have available on predictive tools that operate during the specification phase and try to predict how errors will be distributed in the final delivered products

SECTION X

TASK 11 COST RECOVERY IMPLEMENTATION PLAN

10.1 Cost Recovery

In preparation for the DACS transition from a pilot center to a DoD IAC with the accompanying DoD mandate to recover a portion of operating costs, an extensive study of cost recovery options was undertaken. The study included an extensive literature search, detailed computations of the cost (personnel, computer, etc.) required to produce each DACS product, interviews with the marketing personnel in two other DoD IACs, and a careful examination of responses to the DACS user survey and of DACS interactions with its user community.

A cost recovery system implementation plan was prepared based upon the findings of the study. Section X is devoted to a summarization of that plan and a discussion of the rationale underlying each recommendation in the plan.

10.1.1 Cost Recovery System Implementation Plan

The major aspects of this implementation plan will be:

(1) Definition of chargeable products and services, their costs, and schedule of availability

(2) Establishment of a marketing plan

(3) Methodology for processing reimbursement fees

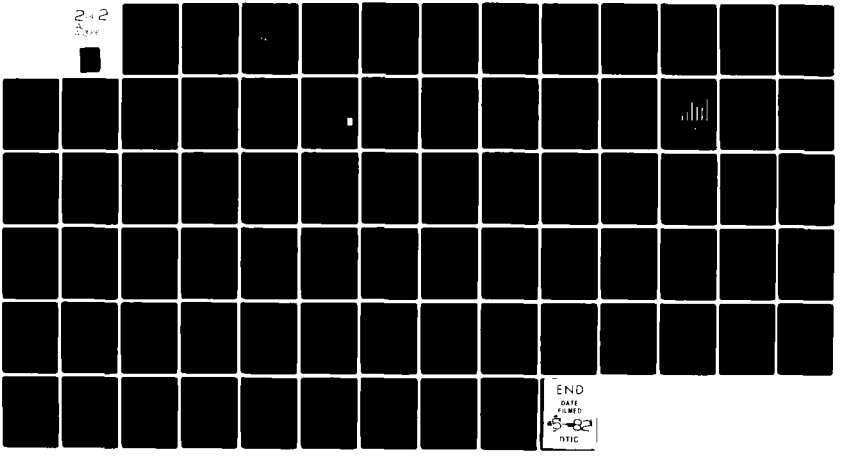
10.1.2 Alternative Pricing Strategies

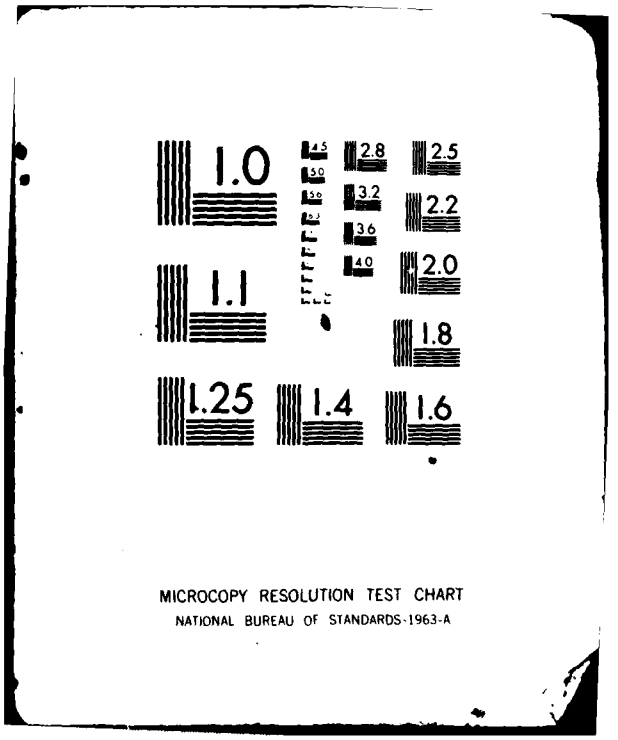
DACS must choose a pricing strategy for each product and service which will be consistent with its sometimes opposing goals of contributing to the public good (i.e. widest possible dissemination of information) and operating in the most cost-effective manner (i.e. recovering 50% or more of base funding). There are several alternatives which can be considered [KING-NCIC], [ZAIS-76], [GE-MTIAC]; those applicable to the DACS are (1) no charge, (2) charge to recover material costs, (3) charge to recover total costs, (4) charge to recover a percentage of total costs (5) charge to optimize revenue, and (6) charge to reflect market value. The most successful IACs, in terms of cost recovery, employ a combination of these alternatives, usually on a product by product basis.

10.2 Setting Prices for Products

Thus, pricing becomes a problem in decision-making which must be part of an overall strategy which has as its end promoting the maximal use of the IAC while meeting the requirements of the cost recovery program. Maximal use of the center is likely to occur when prices are low or non-existent and every user who is aware of the benefits to be obtained

AD-A113 262 IIT RESEARCH INST ROME NY F/G 9/2
ESTABLISHMENT OF THE DATA AND ANALYSIS CENTER FOR SOFTWARE (DAC--ETC(U)
JAN 82 G CARON, D CORNWELL, J DOBMEIER F30602-78-C-0255
UNCLASSIFIED RADC-TR-81-385 NL




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

from using the IAC will avail himself of its services. When prices are set at a high level, the number of users will drop to a lower level and not every user who could make use of the IAC will do so if the cost-benefit analysis with respect to his aims/needs does not yield a positive result. However, a few "right" customers may enable the IAC to sell a customized product which will make a major contribution to the cost recovery program. In practice, both of these have occurred for those IACs most successful in meeting cost recovery guidelines; a judicious mixture of both policies appears in their overall strategy. Our literature survey as well as conversations with GACIAC and RAC personnel indicate that an IAC is unlikely to meet cost recovery guidelines through sales of published documents and custom searches alone, unless the IAC receives these documents in already publishable form, making it necessary to expend only printing, marketing and distribution funds. By far, the greatest contribution to a successful cost recovery program is to be derived from special tasks or studies performed for other governmental agencies, or industrial concerns [GE-MTIAC], [RAC], [GACIAC], [MASON-77], [KING-NCIC]. The next sections contain pricing algorithm recommendations for DACS products and services together with the rationale for the recommendations.

10.3 DACS Products to be Distributed at No Charge

Generally two types of products fall into this classification; those products which are of a promotional nature, i.e., brochures and newsletters, and those products whose development and variable costs are trivial or whose free distribution will promote the use of related products for which the IAC can charge. An example of the latter type of product is the DACS Thesaurus which has been packaged with the BIB-1. The Thesaurus does serve a promotional function in that many users of the bibliographic searches have previously searched the NTIS or DTIC databases and have been unable to effectively deal with the output because the gross level of indexing employed by these services produces an enormous number of citations, the majority of which are irrelevant to the user's information need. Distributing the Thesaurus free of charge enables a prospective user to ascertain for himself the fine level of indexing employed by the DACS and induces some degree of confidence that the DACS can actually deliver the timesavings in locating needed information that it promises.

Those products recommended to be distributed at no charge are the DACS Newsletter, BIB-I and the DACS Thesaurus, the various papers presented by IITRI/DACS personnel at workshops/conferences/symposia and, of course, purely promotional literature.

10.4 DACS Products Priced at Material Cost

Three types of products fall into this classification; (1) those products given to the DACS in a reproducible form, (2) those products of a substantial nature developed by the DACS which have been distributed free during the pilot period and whose distribution is to be continued during the charging period, and (3) those products synthesized from a few sources with modest effort which could help the DACS develop quid pro quo relationships with other agencies or organizations. At present, the

products falling into the first category are the Software Reliability Dataset and the hard copy report which complements it; the data collection forms developed by the Software Engineering Laboratory [BASILI-77] which are being xeroxed and distributed by the DACS and the Nelson report which contains a partial analysis of the DACS productivity dataset. (The RADC productivity dataset logically falls into this category but is being recommended for another classification and will be discussed later). The products falling into the second category are the state-of-the-art reviews "Quantitative Software Models" (SRR-1), and "A Review of Software Maintenance Technology" (SRR-2).

SRR-2 was published by RADC and has been distributed up to the present time by NTIS. It is recommended that SRR-2 be printed and distributed by the DACS to increase DACS visibility in the maintenance area and to broaden the DACS product range at a relatively small cost.

The already-developed products falling into the third category are the various data collection forms developed by DACS personnel relying heavily on a few technical reports or other sources, and short specific monographs on a narrow topic accompanied by an extensive bibliography with abstracts and evaluative annotations, charts, graphs, etc. Another example is the "conversion packet" developed in response to a request for assistance from Gunther AFS, AL. The packet contained a bibliography and an annotated list of specific conversion aids and tools and their distributors, and also a handbook for estimating conversion costs. The DACS then distributed this already-developed packet to several other users faced with a prospective conversion and a lack of information on the subject of tools, costs, or, in general, what to expect.

10.5 Setting Prices to Recover Total Cost

This alternative is also referred to as average cost pricing. In general, the equation to be used is:

(total cost) ÷ (number of copies of the product sold) = (cost per copy)

The difficulty in utilizing this method lies in the difficulty of predicting the shape of the price/demand curve. A hypothetical curve is shown in Figure 10-1. (For information products, the curve does not tend to be linear). In general, as the price of a product increases, the demand for it will decrease. The product of the ordinate and abscissa (represented by the crosshatched area) for any given point on the curve will represent the income which can be derived from selling a given number of the product at a given price. The breakeven point is that point on the curve for which income equals total costs. There may be more than one breakeven point on the curve for a given product. In the case of those information products which have high fixed costs, such as ones which require extensive database compilation and development, the price/demand curve may not even contain one breakeven point. At any rate, in the absence of prior knowledge of the price/demand curve for a given information product, one must make "guesstimates" of the number of copies of the product which can be sold with the stipulation that one not price oneself out of the market.

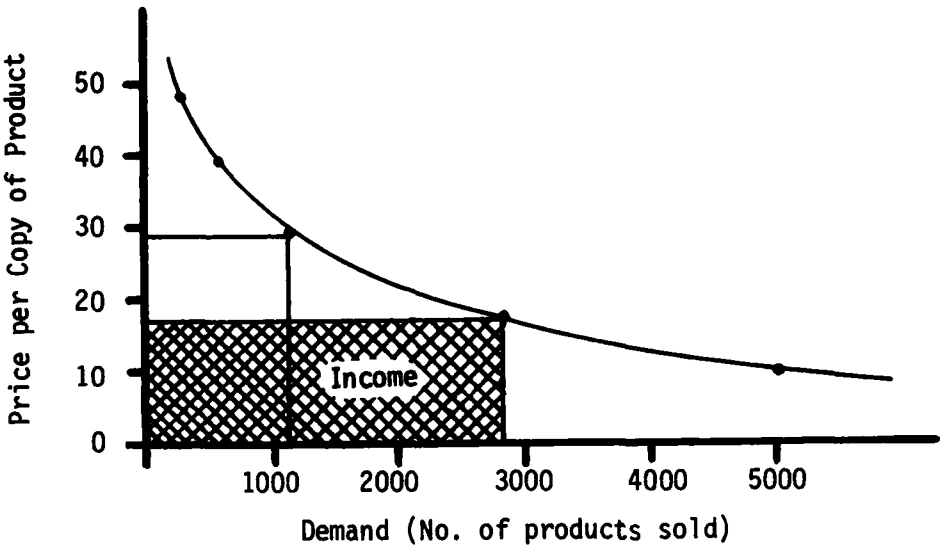


FIGURE 10-1

HYPOTHETICAL PRICE DEMAND CURVE

Those instances where the DACS should definitely recover total costs are not in the distribution of information products directed to the general user community, but for special studies and consulting tasks undertaken for a single user, whether governmental or commercial.

None of the federal agencies whose pricing policies were reported upon in [KING-NCIC], [ZAIS-76], and [MASON-77] tried to recover costs of generating the primary information they distributed. What they did attempt to recover was total costs of processing the information to make it palatable for the user. In the case of the DACS, such a policy would mean that, for the data acquisition program, the costs incurred in locating, negotiating for, and actually acquiring the data would not be targeted for recovery. However, the cost involved in loading the data into databases (exclusive of initial database design and establishment costs), writing the routines to extract a particular subset of the database at a user's request, writing and shipping tapes plus any other professional, clerical, or computer costs involved in meeting a request for a particular access to the database should be recovered. This variation on the total costs recovery strategy could be termed the "recover total costs minus startup costs strategy".

This pricing strategy is recommended for the datasets DACS acquires for which extensive manipulation is required in order to make the data useful to a particular researcher. This type of data will be of interest to only a small portion of the software engineering community. In addition, it is difficult for an individual researcher to acquire such data and put it into a form suitable for analysis. Thus, a dataset readied for analysis should command a premium price. This pricing strategy is recommended for the NASA/SEL dataset and the improved, expanded version of the DACS productivity dataset. The first, NASA/SEL, is nearly unique with respect to the quality of data contained in it; and the second, the DACS productivity dataset, originally compiled by Richard Nelson of RADC, is unique in the range and number of projects it contains. These two factors, uniqueness and difficulty of obtaining elsewhere, coupled with the small number of researchers who would need to use the raw data, argues that demand would be relatively insensitive to price. During 1980, there were 93 requests for either tape or hard copy listings of the DACS Productivity Dataset. Assuming the worst case, a 76% drop in requests, (based upon the worst experiences of other IACs which moved from a free basis to a charging basis) there should be 23 users per year willing to pay for a copy of the improved version of that dataset. The total cost for removing anomalies, incrementing the database with those data elements which it requires and which are contained in other DACS datasets and with new data DACS acquires specifically for that database should be divided by 23 or 46 (one year or two year amortization), variable costs for meeting a specific request added, and that price be set for copies of the raw data in either hard copy or magnetic tape format.

10.6 Setting Prices to Recover a Percentage of Total Costs

Attempting to use this alternative encounters the same difficulty as would be encountered in pricing to recover total costs - the shape of the

price/demand curve is not known beforehand. The only real difference is the decreased risk involved in setting prices to meet goals. Aiming for recovery of 50% of total costs allows the setting of lower per-unit prices, which would probably result in larger demand and the increased social benefit obtained by wider distribution of the technological information contained in the products.

This pricing alternative is not recommended for application to individual products or services of the DACS. It is recommended as a composite goal for the combined products and services offered by the DACS. Certain products, i.e., those which are tertiary information will be distributed free. Other products will be developed with the foreknowledge that their costs are not totally recoverable in any case, but which are developed anyway because of national interests (i.e., enhancement of the U.S. state-of-the-art in software technology may contribute to the economic strength of the U.S.). These products will be counterbalanced by products which recover total costs or more than total cost with the aim of recovering a given percentage (currently 50%) of base funding for the combined product/service mix.

10.7 Charging to Optimize Revenue

This is the pricing strategy employed by commercial firms whose charter stipulates that they are in business to make a profit. This strategy is not appropriate for the centers operated for or by the Federal Government because their aim is not to maximize, or even to make, profits. Indeed, the making of profit is counterproductive to the real aim of the DACS as an IAC, which is the widest possible dissemination of software engineering technology. This is true because the pricing of products to make a profit could ensure that some users who could benefit from the services of the IAC would not be financially able to avail themselves of those services.

10.8 Setting Prices to Reflect Market Value

This pricing strategy is simply to charge for the fair market value of a particular product. The actual price is set in terms of the worth of the information to its prospective user community and by what other IACs charge for comparable information products.

In practice, this strategy is not simple to implement. Substantial experience in the marketplace is required to be able to assess the worth of an IAC's products to its user community. Experiences with user demand during the free period do not translate to the charging period in a linear fashion. There is no way of assessing what proportion of an IAC's users are frivolous users (i.e., users who request products or services out of curiosity rather than need), casual users (users who will make use of the products as long as they are free), and serious users (users who really need the information products of the IAC and are willing to pay for value received) in advance of the imposition of user charges.

It is, therefore, recommended that the pricing decision for the group

of products falling into this category be based primarily on what other IACs charge for similar products until the DACS acquires the requisite experience in the marketplace.

This strategy should be used to set prices for those DACS products and services which bear the strongest resemblance to "standard" IAC products. Products falling into this category are the bibliographic searches, the annotated bibliographies, the second version of the DACS GLOSSARY, the searches and reports which will be produced from the Software Research Projects Database, the data compendia, the handbooks and technical monographs produced as a result of analysis of the database, and future state-of-the-art reports produced by the DACS. In advance of setting prices for DACS products falling into this category, it will be necessary to obtain product listings with descriptions and prices from other IACs, compare specific DACS products with descriptions of the same type of product, and set prices accordingly. Adjustments will have to be made as time goes on if demand at the price set proves excessive or deficient in comparison with the results of periodic user surveys.

10.9 Participation Plans

IITRI's review of the operations and practices of other DoD IACs suggest that participation or subscription plans are a convenient way of inducing users to make use of the IACs products and services on a continuing basis.

The practice of offering participation/subscription plans can serve a dual purpose. It allows the user nearly instant access to the IAC's facilities without the necessity of initiating a purchase order for each use of the center while also tending to promote more frequent use of the IAC. The subscription plan also allows the center to plan for maintenance or expansion of center services with the foreknowledge that there is a given floor of user income to support product and service development.

It is recommended that the DACS offer three levels of participation plans. The first level, offered at a very modest price, would function as as introductory package to DACS products and services. The second level of participation plan would be designed for the frequent user of DACS products and services and would tend to be utilized by corporate or governmental users who have a heavy investment in interest areas addressed by the DACS. The third level would be designed to facilitate the initiation of special studies and tasks which will provide higher levels of income for the DACS and may produce as by-products, services or products which can be tailored and offered to the DACS user community as a whole. An example of this type of task is the transfer of the NBS tools database to the DACS which was funded by a MIPR to the DACS contract. Figure 10-2 is a preliminary version of a brochure which could be used to introduce DACS users to its participation plan.

10.10 Exceptions to Pricing Policies for DACS Products

The overriding concern in establishing prices must be making the DACS

DACS RADC/ISISI
Griffiss AFB, NY 13441
315/330-0937 Autovon 587-3395
Data & Analysis Center for Software

PARTICIPATION PLAN August 1981

Thank you for your interest in the products and services of the Data & Analysis Center for Software (DACS). For a very nominal fee, you can become a subscriber to DACS and take advantage of the various functions, products and services available from the Center. Details and points of contact are described in the DACS User's Guide.

Because timeliness of response often determines the value of information services, DACS offers a variety of participation plans. By enrolling in a participation plan, a member may access DACS resources as needed without the delay of issuing individual purchase requests. A telephone call can begin the processing of a service request and turnaround may be as short as one day, depending upon the nature of the request. Three plans are currently being offered, enabling users to select the service level that best satisfies their particular needs. These plans will be modified as required to reflect the expansion of future DACS products and services.

The minimum service charge category is $tbd per year. It provides four man-hours of DACS engineering staff time for technical inquiries, bibliographic searches or visitations, one tape or hard-copy dataset, copies of periodic current awareness DACS Bulletins, and one copy of the first Annual DACS bibliography (Abstracts and Indices). All services are provided by professional engineering personnel versed in the various disciplines related to software engineering and are supported by the DACS bibliographic database, which consists of bibliographic data entries, indexed and cross-referenced by key word identifiers.

The $tbd service charge category provides the basic products and services of the minimum category plus an additional 20 manhours of professional consultation or equivalent services, and one copy of all regular documents issued by DACS during the participation period (one year). These will include State-of-the-Art Reports, Technology Assessments, Special Study Reports,[1] Annual Bibliography Supplements, and Prepared Bibliographies.[2]

This plan also makes additional copies of all regular DACS publications available at a 20% discount from the list price to all employees of a single corporate entity located at a single plant location.

- -

DACS
RADC/ISISI
Griffiss AFB, NY 13441

Dear Sirs:

I would like to become a DACS subscriber as defined by your participation plan dated August 1981. My check/purchase order in the amount of (circle one): $___ or $___ is enclosed.

Please complete the mailing address information on the reverse side of this form and make check/purchase order payable to: IIT Research Institute (DACS).

The Data & Analysis Center for Software is operated for the Rome Air Development Center by IIT Research Institute

FIGURE 10-2

PARTICIPATION PLAN BROCHURE

The final participation category is open for negotiation and is intended for users that required a more intensive use of DACS capabilities in terms of technical inquiries, bibliographic searches, special studies or mutliple copies of publications. DACS can draw upon the entire technical staff of IIT Research Institute in its normal course of operation and can offer technical expertise in all scientific areas. A negotiated participation plan could include specific assignments of personnel for specialized studies as well as tailored sets of documents or special selected data dissemination services.

The form below is for the convenience of those who are interested in becoming DACS subscribers under the $tbd or $tbd categories. If you are interested in one of these participation plans, please complete the form and return it with your payment or purchase order made out to: IIT Research Institute (DACS). The mailing address is: DACS, RADC/ISISI, Griffiss AFB, NY 13441.

If you are interested in the specialized services described in the final participation category, please contact DACS by letter or telephone to arrange further discussions. Mail such inquiries to: DACS, RADC/ISISI, Griffiss AFB, NY 13441. If you prefer to telephone, contact Shirley Gloss-Soler at (315) 336-0937 or (315) 330-3395.

---

[1]These are reports emanating from special study efforts undertaken by DACS and will be announced in the DACS Newsletter.

[2]These bibliographies result from multiple user inquiries on a single topic and will be announced in the DACS Newsletter. They will be disseminated only upon request.

- -

Name _____ Title _____

Organization _____

Division _____ Mail Stop _____

Street Address _____

City _____

State _____ Zip Code _____

Telephone - Area Code _____ Number _____

Mailing Address (if different from above) _____

FIGURE 10-2 Cont'd

information products available to the maximum number of persons who can utilize their contents while meeting cost recovery guidelines. Thus, if enough special studies and consulting work can be obtained to meet or nearly meet cost recovery guidelines, it is suggested that prices for products which tend to have the most sensitive price/demand curves be lowered to maximize demand and by extension maximize technology transfer in the software engineering community.

In addition to price/demand considerations, the DACS needs a means of rewarding those users who contribute data or other valuable information to the DACS. It is recommended that quid pro quo relationships be established with such users. In return for contribution of data, an agreed-upon level of services would be provided free of charge to such contributors.

10.11 Promotional Program

Once the products have been produced, costs tabulated, and prices set, it is necessary to actively bring them to the attention of the users whose information needs they were designed to satisfy. Not only must users be made aware of the existence of the products and services but they must also be educated in the benefits to be gained through their use.

The promotional mechanisms, brochures, newsletters, advertisements, etc. must be carefully prepared with both the informational needs and the information seeking habits of the end user of the product taken into account.

10.11.1 Promotion to Current Users

The DACS Newsletter will be used as a regular promotional device; in addition to critiques, conference announcements, book reviews, and state-of-the-art surveys, each newsletter will contain a concise listing and description of the products and services offered by the DACS. In addition, new DACS products will be introduced in the DACS Newsletter by a feature article.

Brochures describing newly offered DACS products or services, incorporating sample pages where appropriate, will be prepared and mailed to those individuals and organizations on the Newsletter mailing list.

It should be noted at this point that the DACS Newsletter mailing list was originally established by purchasing one mailing list (the membership of ACM SIGSOFT) and borrowing another (the RADC/OC mailing list). Every individual or organization on either of these two lists was sent a copy of the first DACS Newsletter (Oct 1978) with instructions to fill out a form and return it to the DACS if they wished to receive future issues of the Newsletter. 4,200 copies of the 1978 Newsletter containing the forms were distributed; the forms continued to be returned over several months. By February 1979, the Newsletter mailing list contained 2,049 names. After February 1979, the number of names on the mailing list continued to grow slowly and at this writing, August 1981, it contains 2,735 names. Most of the persons who were added to the mailing list after the original

solicitation learned about the DACS from a colleague or through one of the 22 presentations made at conferences and workshops by DACS personnel.

10.11.2 Promotion to New Users

10.11.2.1 Identification of Demographics of Expected User Population

It will be necessary for the DACS to significantly expand its user base if it is to operate in a cost effective manner. It is highly unlikely that all potential users of the DACS have been identified and are on the mailing list for its newsletter. Thus, a determination must be made as to how best to reach those as yet uncontacted potential users. A detailed study was produced by Forecasting International Ltd. (FI), for the National Science Foundation [BISHOP-77] whose results are directly applicable to this problem. The study surveyed 402 chemists and electrical engineers as to what their scientific and technical information (STI) needs were and how they tended to satisfy them.

The FI team developed a questionnaire based on the market research technique of market segmentation or clustering. The questionnaire was completed by personal interviews with a scientifically designed, nationwide sample of 201 electrical engineers and 201 chemists chosen from the membership lists of the American Chemical Society and Institute of Electrical and Electronics Engineers.

Results of the data analysis for the electrical engineers were presented separately from results of the data analysis for the chemists. Intuition suggests that the results for electrical engineers would provide a fairly close fit to results which could be obtained by a study of software engineering STI market.

The electrical engineers were grouped by FI into six market segments based upon both the type of STI they most frequently sought and their most frequently utilized means of securing such information.

10.11.2.2 Promoting the DACS in Professional Journals

For five of six segments of the electrical engineers, the most often utilized source of STI is professional journals. DACS will place advertisements, press releases and announcements in journals widely read by software engineers. Either the DACS as a whole or specific products will be featured. The advertisements will also contain a coupon to be returned for further information on a given product or a free subscription to the DACS Newsletter and/or a DACS information packet. Coupons will be coded so that the particular journal which prompted user inquiry can be identified. The following are suggested candidate journals to be used as a medium for publicizing the DACS:

Computer (IEEE)

Transactions on Software Engineering (IEEE)

Software Engineering Notes (ACM SIGSOFT)

Communications of the ACM

ISPA Whisper (for data compendia)

Computerworld

Datamation

Transactions on Reliability (IEEE) (joint with RAC)

When tabulation of returned inquiries has identified the journal(s) which provide the most cost-effective means of reaching the potential DACS user community, a small (1/4 page) advertisement will be placed in that journal on a regular basis. Conversations with the RAC and GACIAC marketing personnel indicate that promotion must be continued on a regular basis so that users are reminded that the IAC continues to operate and continues to develop new products. The RAC and GACIAC experiences also indicate that advertisements in professional journals need not be large or expensive (i.e., less than a full page) to be effective, but that they must appear on a regular basis once the most effective journal or journals have been identified.

10.11.2.3 Promoting the DACS Using Free Periodicals

Four of the six market segments identified by FI use free periodicals as a source of STI. The most heavily utilized free periodical for promoting DACS products will be the DACS Newsletter which has been discussed previously. In addition, there are several newspapers and magazines circulated free to software engineers which will print press releases and new product information at no charge to the producer. These will be utilized as new DACS products are developed. An initial list of the free periodicals to which press releases and new product announcements will be sent follows:

OEM News

Systems Engineering News

Datamation

Electronic Design

Computer Decisions

This list will be revised as experience indicates which are worthwhile.

10.11.2.4 Promoting the DACS Through Librarians

The Forecasting International survey identified librarians as both

users of STI and a means of reaching other STI users. Indeed, their survey found that the central librarian "plays a dominant role in requesting the purchase of STI sources and appears to be the chief purchase intermediary". [BISHOP-77] Certain DACS products, especially the Annotated Bibliographies and SOARS should be marketed to librarians. There are three broad classes of librarians to be reached. The first, corporate librarians, can best be reached through the monthly journal, Special Libraries, published by the Special Libraries Association. The second, governmental agency librarians, can be reached by the inclusion of the DACS in the DoD publication, "Information Analysis Centers Profiles for Specialized Technical Information" which receives wide distribution to DoD and governmental agencies. The first issue which featured the DACS was distributed in April 1981. As of this writing, the DACS has already received several service requests referencing that booklet. The third is the specialty and departmental librarians at major colleges and universities, especially those attached to the engineering schools. These can be contacted by use of the Computing Newsletter edited by Daniel Cougar at the University of Colorado at Boulder and distributed to college and university departments of computer science worldwide.

10.12 Summary of Cost Recovery Plan

The DACS has distilled the information contained in several studies and reports as well as the experiences of the two other IACS (RAC, GACIAC) to develop a program which will meet the cost recovery goals specified for the DACS. The program in addition to the rationale summarized here contains recommended costing algorithms for products and services and suggested reimbursement procedures. After implementation of the cost recovery program, its effectiveness will be assessed on a continuing basis and adjustments made as required to operate the program in the most cost effective manner possible.

SECTION XI

SPECIAL TASKS

11.1 Introduction

During the last year the DACS has performed two special tasks which were funded above and beyond the original contract. This section will give a general description of those efforts.

11.2 Install NBS Software Tools Database

11.2.1 Background

The increased use of software tools is considered to be one of the most promising approaches to alleviation of the related problems of high software cost and low productivity. This increased use is dependent upon the accessibility of information about tools to potential users. The U.S. National Bureau of Standards (NBS) developed a software tools database which contains informaion on 250 software tools. NBS could no longer support this database and was seeking a new facility which could support its tools database and make it available to the software engineering community. The information in such a database of tools could be used by management to develop software engineering environments and/or methodologies.

11.2.2 Problem

The purpose of this effort was to enhance the services offered by the DACS through the installation of the NBS Software Tools Database as well as to make this database available to AFLC/NSW users. The effort, therefore, involved the transfer of the NBS Software Tools Database from its host system at NBS, a DEC 10 using PASCAL/R, to an RADC computer facility. Once the specifications and requirements were identified, an evaluation was made to determine which of the RADC computer facilities would best host the database providing all the capabilities that it possessed at the NBS facility.

This task involved identifying potential target systems to host the database. This selection was based on two fundamental requirements.

(1) The system had to be accessible to the AFLC/NSW users specifically; the software community in general, and therefore, must be on the Arpanet.

(2) The system should provide user friendly information retrieval on tool features, languages, developers, documentation, hardware and software requirements, availability, publications and contracts.

The two RADC computer systems that were selected as potential target systems were:

- DEC 20 - TOPS 20
- Honeywell 6180 - MULTICS

Each of these systems will be discussed with respect to their capability to meet the requirements stated above.

11.2.2.1 DEC 20

This system does not currently support a database management system, and in particular does not support PASCAL/R. However, it was selected as a potential target system since it met the first of the previously mentioned requirements. Additionally, the system has an automatic DEC 10 emulation capability within the DEC 20 processor. Therefore, it was thought that transferring the relational database and compiled procedures could quickly provide a replication of the data and information retrieval capabilities hosted at NBS. If an evaluation of this system determined that it best fulfilled the requirements stated previously then the acquisition of PASCAL/R, for the DEC 20, would be pursued. This would then provide the necessary facility to modify, enhance and add to the existing system.

11.2.2.2 Results of Investigation of DEC 20

An initial visit was made to the NBS computer system site. The specifications and requirements of the NBS Software Tools Database were reviewed. NBS personnel demonstrated the contents and capabilities of their system. A query was selected as the initial software procedure to be compiled on the DEC 10 and transferred along with the accompaning database to the DEC 20 to be executed in emulation mode. Initial transfer was made via magnetic tape media.

The compiled object code was loaded into the DEC 20 but failed to execute successfully.

It was initially conveyed to us that part of the problems encountered were due to the incompatibility of the processors on the DEC 10 and DEC 20 systems. Several unsuccessful attempts were made to overcome these difficulties however the incompatibilities between the two machines were not resolved.

To overcome these problems it would be necessary to procure a PASCAL/R compiler modified for the RADC DEC 20 and recompile the existing software and procedures. Additionally, the database might have to be reloaded to be compatible with the structures created by the new compiler. This information was relayed to the cognizant RADC engineer for review and determination of interest in pursuing this route.

11.2.2.3 Honeywell 6180 - Multics

Recognizing that the approach selected for the DEC 20 might present some difficulty, and since the alloted time period was short, a parallel

effort was undertaken to better assure some degree of success.

An implementation under Multics using Multics Relational Data Store (MRDS) and Linus query language was undertaken. All software had to be developed to install the database and to replicate the capabilities at NBS.

Additionally, an evaluation comparing the two systems, DEC 20 with PASCAL/R and the Honeywell 6180 with MRDS was made and a recommendation of which system will best host the database was made.

11.2.2.4 Results of Investigation of Multics

Due to time constraints, data for only 25 (of the 250) tools were installed. This approach was pursued so that the capabilities and limitations of MRDS could be explored and exercised against a subset of the database. The details of this work are contained in a separate publication entitled "NBS Tool Database System Description and User's Manual."

11.2.2.5 Conclusions

In general, the Honeywell 6180 using MRDS proved to be a satisfactory host for the NBS Software Tools Database. The hardcopy publication format of "NBS Software Tools Database" produced by NBS was replicated. Queries were written in PL/I, FORTRAN and in a query language, Linus. Thus, the software community would have varied choices of access. Most of the limitations that exist under MRDS and which are addressed in the manual referenced above also exist in PASCAL/R. Some additional capability such as easy boolean retrievals, quick response via ad hoc queries through MRDS system calls, flexibility of database access via a data manipulation language from any HOL language and a generally more user friendly interface to the database appear superior to PASCAL/R. PASCAL/R does not have a query language easily used by the non-programmer as does MRDS (Linus).

Also, it is important to note that MRDS is commercially supported by Honeywell. This is not true of PASCAL/R. Support for this compiler is provided on an informal basis by the University of Hamburg.

11.3 NSW Tools Training Project

11.3.1 Background

The increased use of software tools is considered to be one of the promising approaches to alleviation of the dual problems of high software cost and low productivity. This increased use is dependent upon (1) the accessibility of information about tools to potential users, and, (2) the training of users on the utilization of the tools. This task is related to the second need.

With the increasing tendency of governmental agencies to make software tools accessible to general users via networking facilities a corresponding need arises to acquaint those users with the capabilities of the tool and

the mechanisms and procedures necessary to properly use the tool.

11.3.2 Problem

The purpose of this effort was to survey and evaluate the tool training techniques presently available either commercially or through government sources and to recommend the appropriate training scenarios for the use of software tools on the National Software Works (NSW). This was followed with the selection of an NSW tool and detailing the training approach for this tool as recommended by this study as well as a plan for its implementation and for incorporating additional training aids into the NSW.

11.3.3 Summary of Study

In support of this effort this task produced a document entitled "Scenario for Tool Training on the NSW" which contains a definition of available training techniques. It was thought that the training in the use of these software tools would best be made available in the same way the tool is made available, via the facilities of the network utilizing online instruction. Therefore, this report summarizes only the characteristics of available computer based training materials. It explores computer aided instruction (CAI) in general, illustrates characteristics and features, and evaluates the useability. Similar summaries are not provided for other media i.e., non-computer related methodologies.

The various approaches using different simplicity/complexity levels are discussed. The major advantages and deficiencies are outlined and the recommended approach for the NSW tool training is made. Each is evaluated with respect to the requirements of the AFLC/NSW users. The selection of the tool training approach considered such factors as cost, ease of implementation, resources required and finally a profile of the user of these training aids.

The assessment of these factors was supported by literature searches, interviews with tool users and IITRI's own experience in training personnel in tool usage. The results of this analysis culminated in the establishment of a general framework for training scenarios and the initial selection of a tool for which this tool training capability will be developed. Finally, a roadmap for the implementation of this tool training was provided.

SECTION XII

TASK 9 AND 10 CENTER EVALUATION AND EFFECTIVENESS

12.1 Task 9 Center Evaluation Mechanisms

12.1.1 Introduction

The purpose of this task is to develop a philosophy and criteria for evaluating the effectiveness of the DACS. There are essentially two approaches to determining the needs of current DACS users; (1) ask them what they want and (2) tabulate on a historical basis what they have requested. IITRI, during its operation of the DACS, has done both and each will be discussed here. The first approach took the form of a user survey; the second approach took the form of classifying and tabulating the technical inquiries and bibliographic searches on a topical basis.

The first approach is discussed in this section, the second approach is discussed in Section 12.2, Improving Center Effectiveness.

12.1.2 The DACS User Survey

12.1.2.1 Introduction

Two user surveys were performed by the DACS. The results of the first survey are reported in The DACS Interim Report, RADC-TR-80-204. The second was distributed in early July of 1980 and analyzed during August and September. Results were integrated into a DACS internal report [WIT-80]. Major results of the survey are summarized below.

The survey (Figure 12-1) contained 13 questions which were intended to characterize user organizations, job descriptions, and areas of concern in the field of software engineering. Some questions explored users' information seeking and sharing behavior and their motivation to consult DACS. Products were evaluated through ranking and by examining the benefits of a particular user. One free-response question elicited suggestions for new products or services, and the last question made it possible for the respondent to receive a free DACS information packet (incentive to return the survey).

As of that writing, 193 responses were received, amounting to 43% of surveys distributed. Two sub-groups, intensive users and casual users, returned 112 and 81 surveys, respectively. Foreign respondents consisted of 8.5% of those who answered.

12.1.2.2 Summary of Results

Results of the survey are reported here for each question or group of questions. Any significant differences between the responses of intensive users and of casual users are noted.

DACS
USER SURVEY

1. Would you classify your organization as:
   - ___ Academic
   - ___ Government or Government Agency
   - ___ Business/Industry
   - ___ Industry under Government contract
   - ___ Other (please specify) _____

2. Which title most closely describes your job?
   - ___ Instructor
   - ___ Manager
   - ___ Systems Analyst
   - ___ Researcher
   - ___ Consultant
   - ___ Programmer
   - ___ Other _____

3. In your computer software work, what are your areas of concern? (check any that apply)
   - ___ Cost Reduction
   - ___ Design, Development, Productivity of Computer Software
   - ___ Software Maintenance
   - ___ Managing Software Projects
   - ___ Monitoring and Controlling Software Quality
   - ___ Research (indicate area) _____
   - ___ Other _____

   Comments: _____

4. Of the DACS Products and services you've used, which are most useful? Please rank, 1 - most useful. For products you haven't used at all, put an X in the space.
   - ___ Newsletter
   - ___ DACS Glossary
   - ___ Database Subsets (e.g. Software Reliability Database)
   - ___ State-of-the-Art Reports (e.g. Quantitative Software Models)
   - ___ Custom Searches (bibliographies)
   - ___ Technical Inquiries (consulting)
   - ___ Other _____

   Comments: _____

IIT Research Institute
Box 1355, Branch PO
Rome, NY 13440

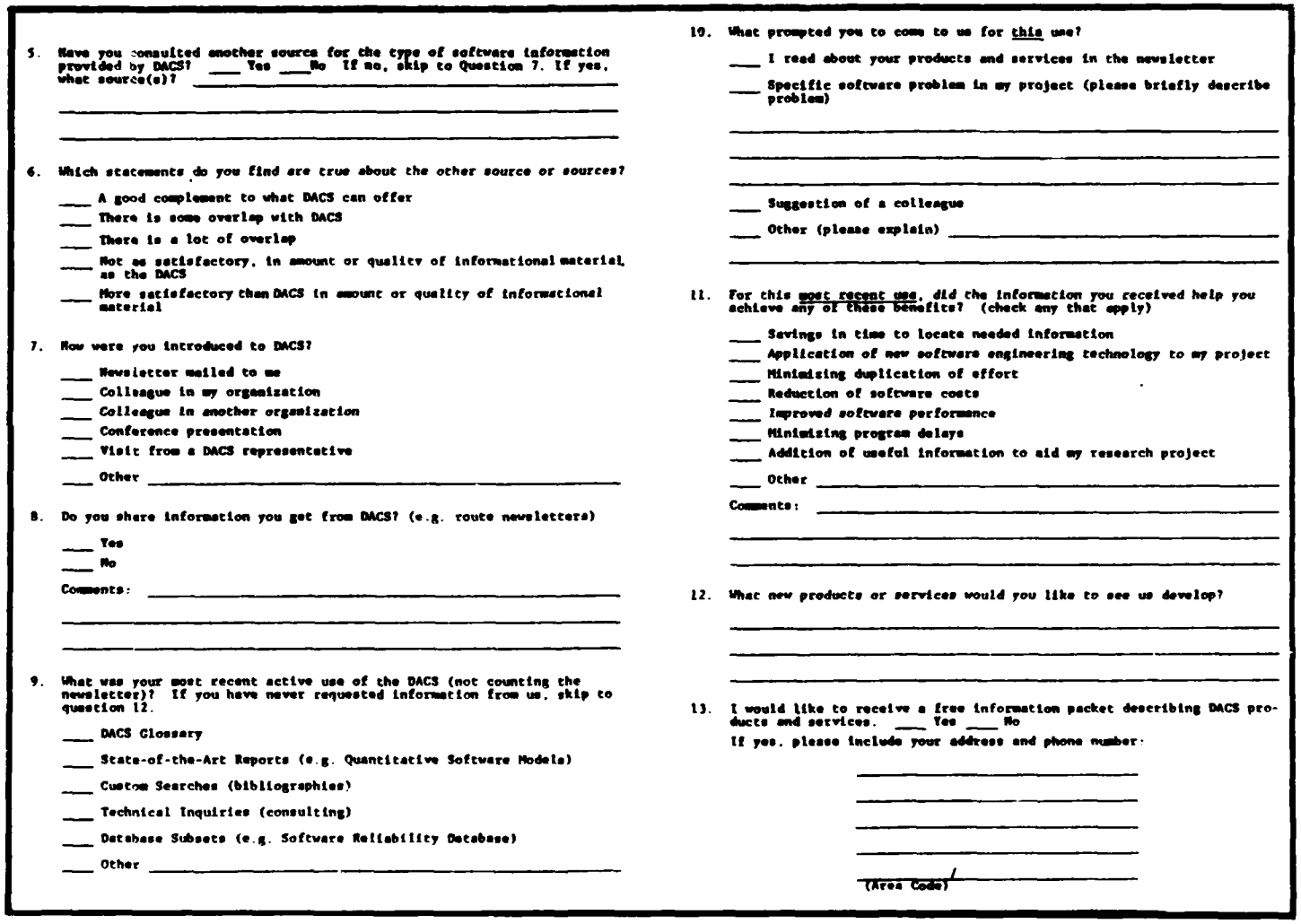
NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 39 ROME, NY 13440
POSTAGE WILL BE PAID BY ADDRESSEE

Susan Witter
IIT Research Institute
Box 1355, Branch P.O.
Rome, NY 13440

FIGURE 12-1

DACS USER SURVEY



5. Have you consulted another source for the type of software information provided by DACS? ___ Yes ___No If no, skip to Question 7. If yes, what source(s)? _____

6. Which statements do you find are true about the other source or sources?

___ A good complement to what DACS can offer
___ There is some overlap with DACS
___ There is a lot of overlap
___ Not as satisfactory, in amount or quality of informational material, as the DACS
___ More satisfactory than DACS in amount or quality of informational material

7. How were you introduced to DACS?

___ Newsletter mailed to me
___ Colleague in my organization
___ Colleague in another organization
___ Conference presentation
___ Visit from a DACS representative
___ Other _____

8. Do you share information you get from DACS? (e.g. route newsletters)

___ Yes
___ No

Comments: _____

9. What was your most recent active use of the DACS (not counting the newsletter)? If you have never requested information from us, skip to question 12.

___ DACS Glossary
___ State-of-the-Art Reports (e.g. Quantitative Software Models)
___ Custom Searches (bibliographies)
___ Technical Inquiries (consulting)
___ Database Subsets (e.g. Software Reliability Database)
___ Other _____

10. What prompted you to come to us for this use?

___ I read about your products and services in the newsletter
___ Specific software problem in my project (please briefly describe problem)
___ Suggestion of a colleague
___ Other (please explain) _____

11. For this most recent use, did the information you received help you achieve any of these benefits? (check any that apply)

___ Savings in time to locate needed information
___ Application of new software engineering technology to my project
___ Minimizing duplication of effort
___ Reduction of software costs
___ Improved software performance
___ Minimizing program delays
___ Addition of useful information to aid my research project
___ Other _____

Comments: _____

12. What new products or services would you like to see us develop?

13. I would like to receive a free information packet describing DACS products and services. ___ Yes ___ No

If yes, please include your address and phone number:

(Area Code)

FIGURE 12-1 Cont'd

Question 1 - Organizational Affiliation

This question was included to determine the working environment and affiliation of DACS users. Figure 12-2 shows the distribution of responses. Combining the responses of those who classified themselves as "government or government agency" together with those who classified themselves as "industry under government contract" leads to the conclusion that 56% of the respondents used DACS for government work.

Question 2 - Type of Job

The user was given seven choices of job type. Figure 12-3 shows the proportions of jobs performed by respondents. It is interesting to note that one group, managers, holds 34% of the total.

Question 3 - Areas of Concern

This question determined what aspects of software engineering were of interest. Users were encouraged to choose as many areas as necessary, therefore, the responses total more than 100%. Figure 12-4 demonstrates the percent of respondents who chose each area of concern.

Question 4 - Products/Services Evaluation

Users were asked to rank products and services according to their usefulness and to indicate which products or services they had not used. Responses are exhibited in Figure 12-5. Those people who had used a product/service tended to rank it highly; products or services which were not rated as highly had been used by less people. This suggests a generally positive opinion of usefulness of DACS products/services. Increased visibility of the less used products should result in their increased application to user needs.

Questions 5 and 6 - Comparison of Other Sources of STI with DACS

Questions 5 and 6 were included in order to determine whether DACS is unique in terms of the scope and quality of its products and services. Another factor can be deduced from these questions: the information seeking behavior of users.

Figure 12-6 demonstrates that fewer casual users consulted other sources than intensive users. In both cases, less than half the respondents did consult other sources for the type of software information available at DACS. For those who did consult other sources besides DACS, the most frequently mentioned sources are in this order:

Technical Libraries

NTIS

IEEE Computer Society

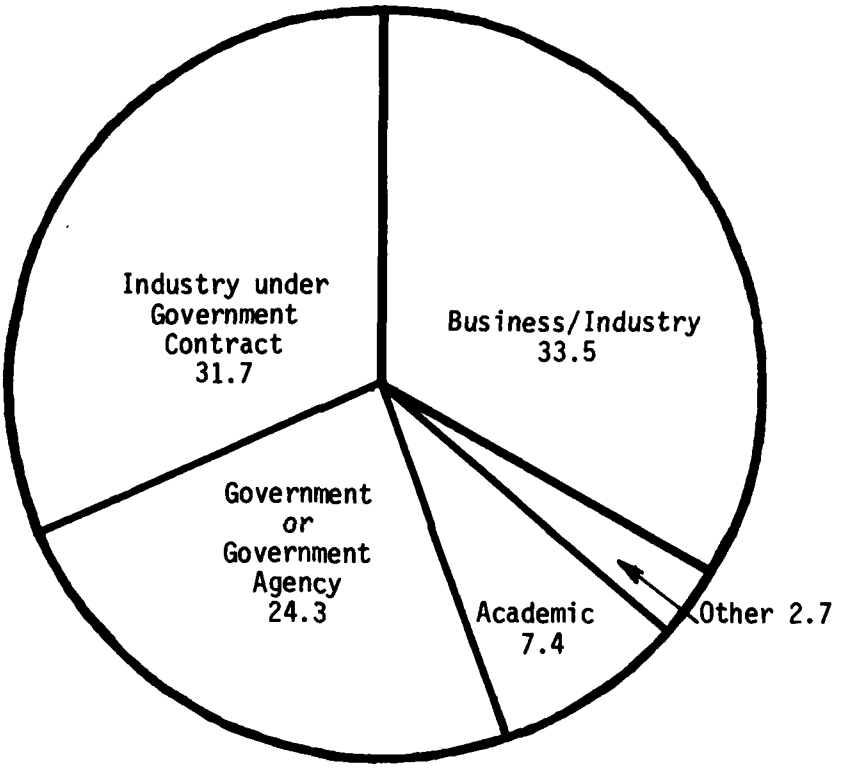
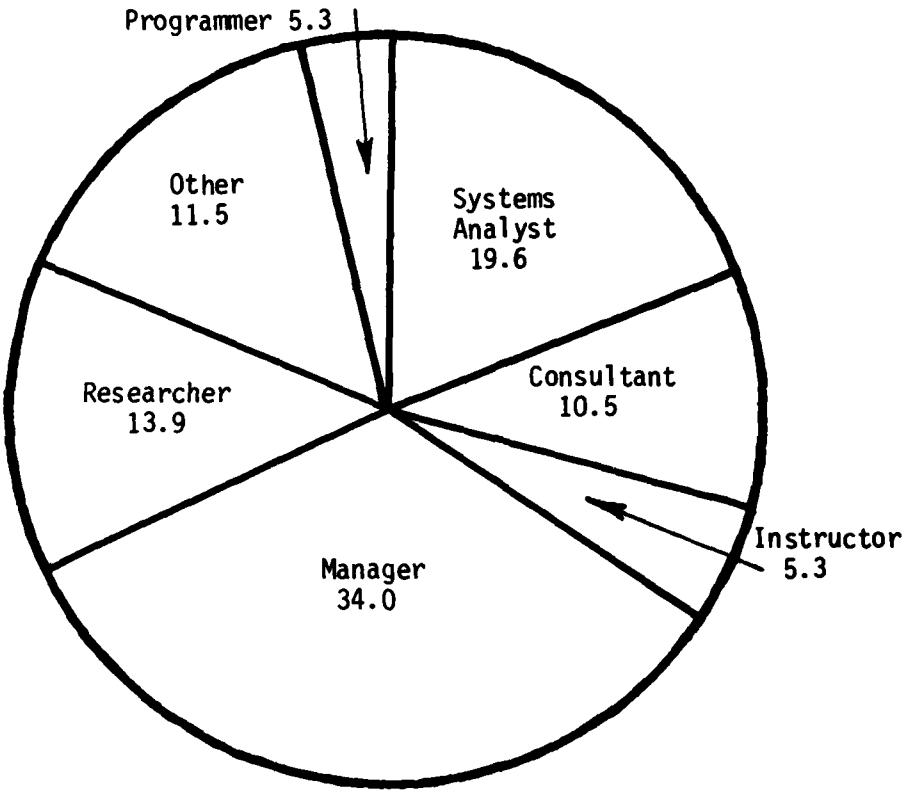


FIGURE 12-2

ORGANIZATIONAL CATEGORIES



FIGUR E 12-3

JOBS

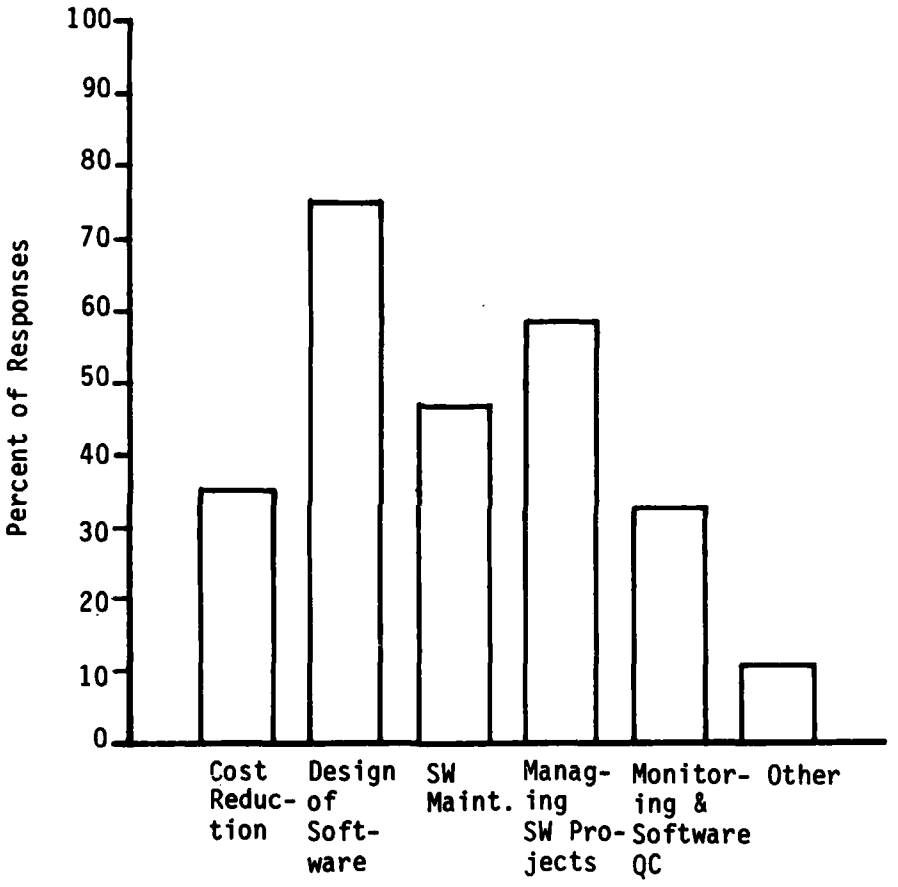


FIGURE 12-4

AREAS OF CONCERN

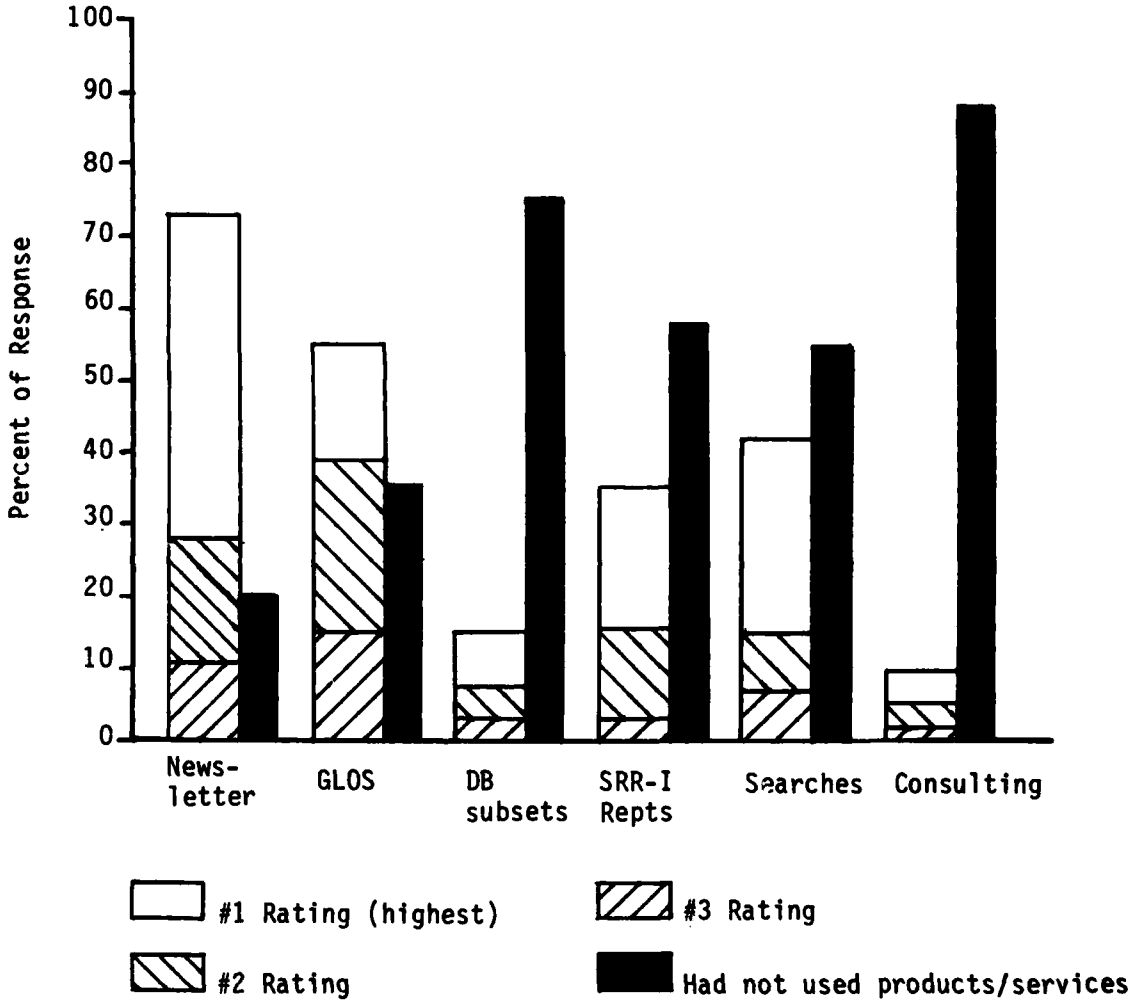


FIGURE 12-5
RATING OF PRODUCTS

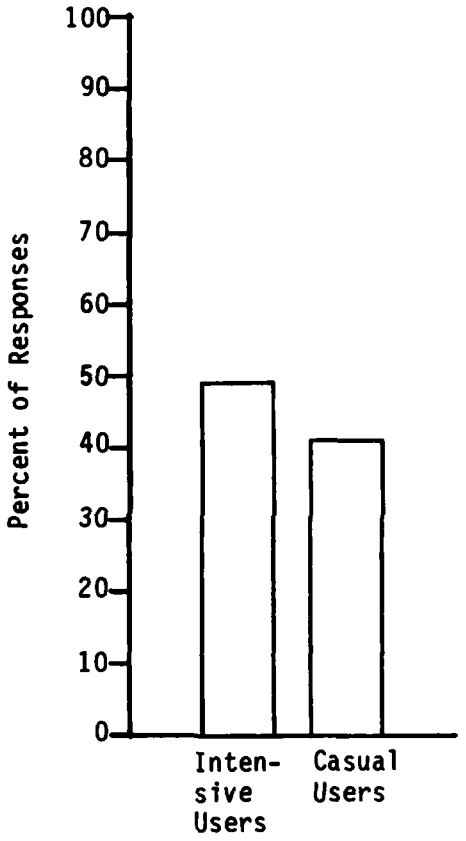


FIGURE 12-6

CONSULT OTHER SOURCES

ACM

Conferences

Colleagues

DTIC

Data Pro Research

Most of these sources are clearinghouses for current literature and research, only one of the functions performed by DACS. A wide variety of other sources including databases, corporations, and government agencies, was also mentioned.

For users who had consulted other sources, question 6 asked for a comparison between these sources and DACS. It can be seen from the partial list of other sources that none are exactly equivalent to DACS in terms of types of services provided.

Here is the breakdown of evaluations of other sources (again, some users chose more than one):

Good complement to DACS:	59.5%
Some overlap with DACS:	19.0%
Lots of overlap with DACS:	3.8%
Not as satisfactory as DACS:	15.8%
More satisfactory than DACS:	2.5%

Questions 7 and 8 - Means of Introduction to the DACS

Questions 7 and 8 were aimed at determining which of the currently employed promotional means were most effective in finding users. An influencing factor is that a mailing list was purchased, from which the DACS Newsletter mailing list was derived. In spite of this, colleagues seem to have been very effective in spreading information about DACS. Figure 12-7 shows the breakdown.

Most of the respondents who chose "other" for this question named the means of introduction as an article in a computing magazine.

The majority of respondents indicated they do share information they receive from the DACS. The proportion was slightly less for casual users (75.0%) than for intensive users (88.1%).

Questions 9, 10 and 11 - Benefits Obtained by Use of DACS Products and Services

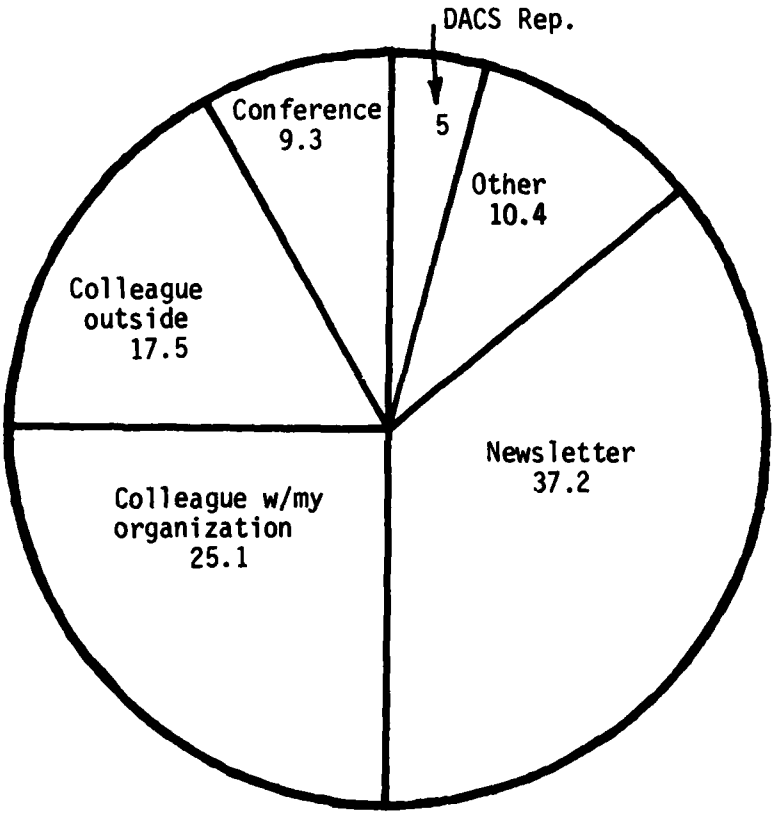


FIGURE 12-7

HOW WERE YOU INTRODUCED TO DACS?

Three questions; 9, 10, and 11; formed a set to explore specific benefits and motivations for a particular use. The intent of the question set was not to compare the benefits of particular products, but to help respondents recall some specific benefits by associating them with recent use.

Those who did complete the three-question set (respondents who had used a product or service) number 85% of all respondents. Their inducement for the most recent use was divided in this way:

Newsletter:	53.5%
Specific Problem:	21.8%
Suggestion of a Colleague:	16.9%
Other:	7.7%

"Other" included conference presentations and former contact with DACS personnel.

Benefits of product/service user (see Figure 12-8) were most heavily concentrated on the finding and utilization of information. Indirect, long-term benefits, such as reduction of software costs, were reported very infrequently. Benefits associated with a particular product or service follow the pattern of benefits for all uses, with the exception that indirect benefits increased slightly for data base subsets. For some products, uses were too few in number to give a reliable distribution of benefits.

Question 12 - New Products and Services Suggested

Each of the following products and services was suggested in the survey by four or more respondents:

Annotated bibliography

List of consultants or contacts

Directory evaluation of software tools

Upgrade searches (include abstracts)

Life cycle cost models and data

Error data

Directory of software programs and services in the Federal Government

Periodic technical workshops, seminars, or courses

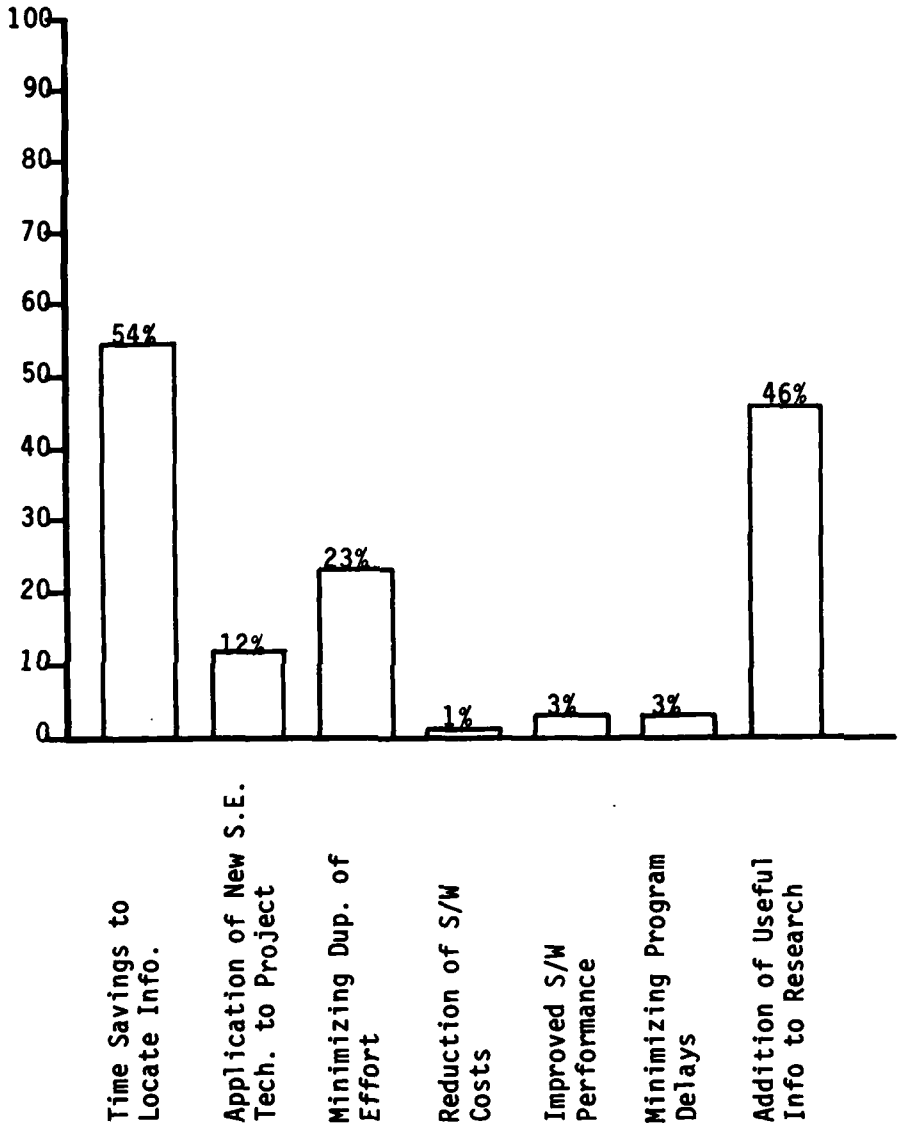


FIGURE 12-8

BENEFITS OF DACS PRODUCTS AND SERVICES

12.2 Task 10 Improving Center Effectiveness

12.2.1 Introduction

This section of the report provides an analysis of the inquiries processed during Phases I, II, and III of the contract period, a discussion on the processing of the documents for the software engineering bibliographic database, and a description of the recommended future activities.

An accounting mechanism has been established which provides the capability to record all resource expenditures (personnel hours and cost, travel costs, and purchased materials and services) by major DACS activity.

This mechanism also allows the tracking of user inquiries. There are 24 major activities (ventures) and they are listed in Figure 12-9. This data is reported on a monthly basis.

Information on the services requested is recorded on the DACS Service Request Record (SVC). This record contains the name and affiliation of the user, the date requested and answered, the request type, a narrative description of the request, the recommended action, the response, and the resources expended. Each SVC is assigned a unique number. A User Profile Record (file) is maintained for each organization and summarizes the requests of all individuals within the organization. The SVC's are then filed in their respective User Profile Folder. A DACS document log is also maintained on a continuing basis to record document distribution according to type, date, and user.

A document processing log provides controls for keeping track of the processing of software engineering documents for the DACS document library. This log records the data the document was assigned a unique document accession number, the date when the document was coded and indexed, the date the coding was verified, the date the coding sheets were sent to be keypunched, and the date when the document citation was entered into the bibliographic database.

This operational data is summarized and incorporated into the monthly status report. The status report contains the number of requests processed, the number and types of products (documents) distributed, the growth of the database, the number of names on the newsletter mailing list, and the number of documents received, accessioned, and entered into the bibliographic data files.

12.2.2 User Inquiries

12.2.2.1 Phase I

During Phase I (15 August 1978 to 1 April 1979) of the development of the Center, 91 technical and information inquiries were received and processed. The majority of these requests centered around more information about the Center. In answer to these inquiries, copies of two papers were

Task 1 Establish and Operate

A 01 Chicago Based Administration and Direction
A 10 Install NBS Software Tools Database
A 11 Software Tool Training Requirements
A 17 Establish and operate the Center - administrative

Tasks 2 and 5 Develop DB and Process Input

A 20 Database development and maintenance
A 21 Input processing - data
A 25 Data Acquisition Program

Tasks 3 and 4 Data Analysis and Data Parameters

A 30 Data Analysis Program (SRR-2 production)
A 31 BSDS Analysis
A 32 Data Parameters

Task 6 Current Awareness Program

A 40 Develop and expand general user awareness (newsletters and promotional material)
A 41 Preparation and participation in meetings and conferences
A 42 Reliability Handbook Workshop

Task 7 STINFO

A 50 Establish and maintain library
A 51 Software Engineering Glossary
A 52 Review research projects

Task 8 Products and Services

A 60 Preparation and distribution of data subsets
A 61 Preparation and distribution of data compendiums
A 62 Final preparation and distribution of technical reports (SRR-1)
A 63 Final preparation and distribution of bibliographic products (BIB-1, Thesaurus, Glossary)
A 64 Technical and bibliographic inquiry services (SVC)

Tasks 9, 10 and 11

A 70 Evaluate and improve Center
A 71 Establish a cost recovery program

Task 12

A 80 Computer program documentation

FIGURE 12-9

DACS ACTIVITIES

distributed entitled "An Introduction to the Data and Analysis Center for Software" and "An Introduction to the DACS Software Engineering Bibliographic Database."

During this period, we were not well equipped to provide much technical assistance; although we did produce, manually, some custom bibliographies and provided referrals to other sources of information. The table below illustrates the number of inquiries processed per month.

MONTH	NUMBER
----------1978----------	
October	2
November	12
December	12
----------1979----------	
January	12
February	20
March	33

These inquiries do not reflect those users who requested to receive the DACS Newsletter. As of 1 April, 1979, there were 2143 names/organizations on the Newsletter mailing list.

12.2.2.2 Phase II

With the onset of Phase II (1 April 1979 to 1 February 1980), we modified our method for recording and processing of user requests. We classified the requests into three basic categories:

- Newsletter Inquiries
- Document Inquiries
- Technical Inquiries

A Newsletter inquiry was a request to be placed on the Newsletter mailing list; a document inquiry was a specific request for a DACS document or documents; a technical inquiry was a request for technical information which required some engineering analysis to respond. These technical inquiries we answered in a variety of ways dependent upon the type of request, the depth and breadth of our information base on the subject area of interest, and the availability of engineering time. Custom bibliographic searches were performed, analyzed, and synthesized; DACS documents, or subsets of the DACS database, were distributed, and/or the user was referred to other sources of information or contacts.

Figure 12-10 provides information on the number of requests for each category per month. The total number of inquiries processed during this

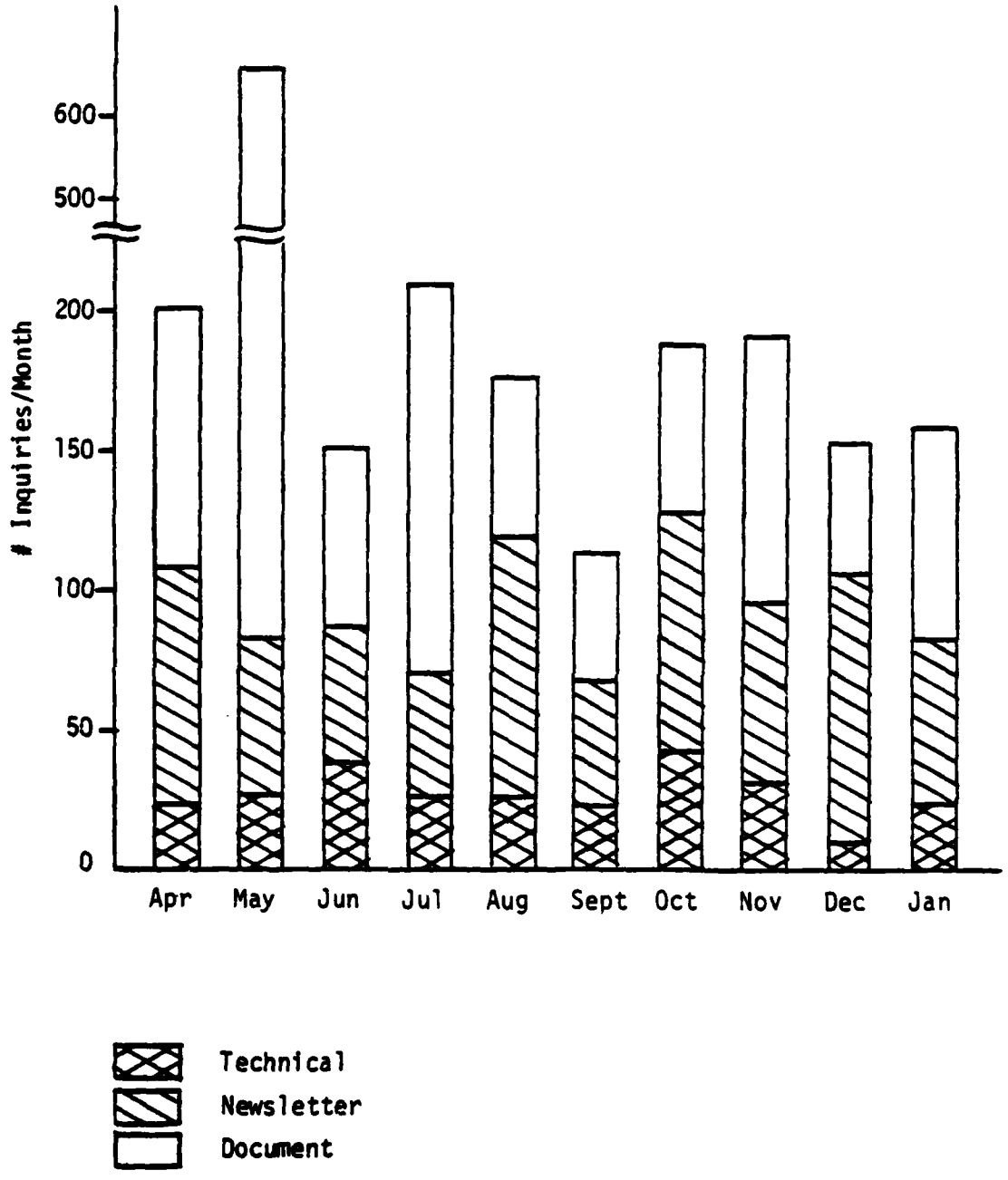


FIGURE 12-10

INQUIRIES/MONTH PHASE II

10-month period (1 April 1979 to 1 February 1980) was 2,162. This averages out to approximately ten inquiries per day. The total number of inquiries by category were:

Technical	281
Newsletter	669
Document	1212
TOTAL	2162

The percentage of technical inquiries by institution type is listed below.

Requestor	Percent of Total Technical Inquiries
Government	32%
Industrial	60%
Academic	4%
Foreign	4%

This data was compared to the percentages from two DLA information analysis centers; Metals and Ceramics Information Analysis Center (MCIC), [RUBIN-77] and the Reliability Analysis Center (RAC), [EHREN-78] .

Requestor	Percent of Total Inquiries	
	RAC	MCIC
Government	19%	20%
Industrial	76%	74%
Academic	(not recorded)	3.5%
Foreign	14%	2.5%

One of the major benefits of an information center is to provide rapid response to requests for technical information. In analyzing the response time data for Phase II, we had not processed requests in a timely enough manner. Steps were taken to alleviate these problems by assigning more flexible responsibilities so that work loads for individual members of the staff could be distributed more evenly.

Below is a summary of the number of working days to answer a request during this ten-month period.

Number of Working Days	Percent of Total Technical Inquiries
0- 4	61%
5- 9	16%
10-14	8%
$\geq$ 15	15%

During December and January approximately 90% were answered within the first five days while in April and November only 20% were answered in the

first five days.

Over 3200 documents were distributed in response to the 1200 document requests. Below is a list of the documents most requested.

Document Title	Number Distributed
Bibliographic Services, Custom Searches (BIB-1)	817
DACS Thesaurus	802
DACS Glossary (GLOS-1)	764
Quantitative Software Models (SRR-1)	764
An Introduction to the Data and Analysis Center for Software	195

The respondees to the first DACS User Survey indicated a preference for state-of-the-art reports. They also indicated a low interest in custom bibliographies. This is somewhat out of line with the statistics on the requests for documents, in that, almost twice as many requests for the bibliographic services guide were received than for the state-of-the-art report on "Quantitative Software Models." This may be explained in part by the fact that this report is in a specialized area.

Another mechanism for receiving feedback from users is to include a critique response form with each document and/or technical inquiry response. However, the response to this mechanism tends to be low, as reported in [CRUM-77]. The Defense Technical Information Center reported on one sample of critique forms sent with 4262 documents. Only 98 of the forms were returned, or 2.3 percent. They indicate that a better rate is achieved only as a result of active follow-up by project personnel.

12.2.2.3 Phase III

In February 1980, the classification used to identify user requests was again modified. The catagories used in this classification are:

- Technical Inquiries
- Document Inquiries
- Dataset Inquiries
- Newsletter Mailing List Inquiries

Dataset inquiries were added to the reporting procedures to more effectively track user interest in these products. A dataset inquiry is defined to be a request for either the hardcopy or machine readable form of any of the datasets available for distribution from DACS.

Figure 12-11 provides information on the number of requests in each catagory per month. The total number of inquiries processed during this 18 month period (1 February 1980 - 1 August 1981) was 2120. This figure does not reflect inquiries made to delete users from the mailing list. This

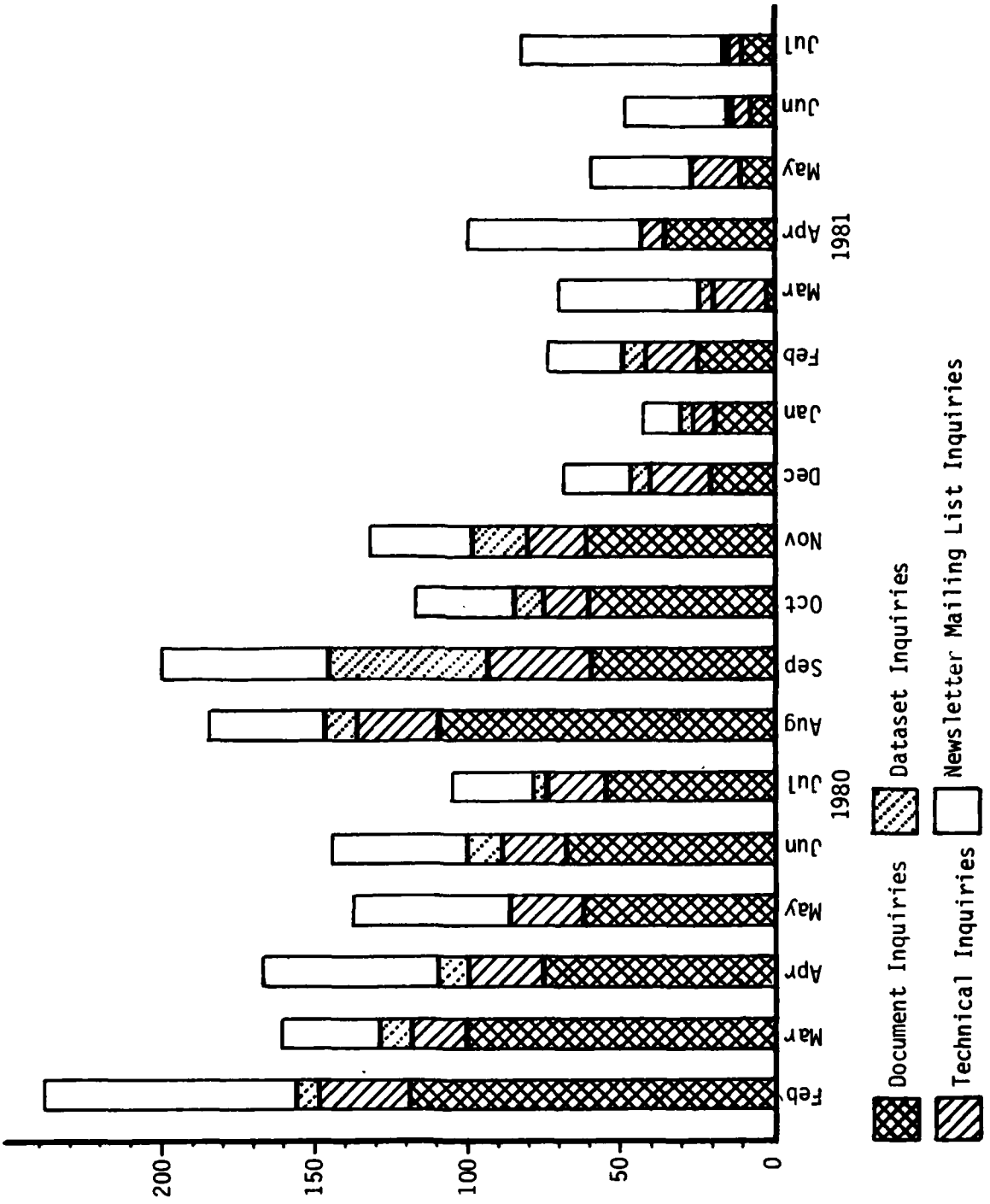


FIGURE 12-11

INQUIRIES/MONTH PHASE III

averages out to almost six inquiries per working day. The total number of inquiries per catagory during this period were:

Technical	325	
Document	900	
Dataset	147	
Newsletter	748	(additions only)
TOTAL	2120	

A number of measures were instituted toward the end of this phase, to reduce the number of users, so that more time could be devoted to data acquisition and analysis activities. The gradual decrease in the number of inquiries per month, as depicted in Figure 12-11, is due to the institution of these measures.

In January 1981 the DACS announced that it was necessary to delay the processing of non DoD user requests. This resulted in a cutback in service to industry customers but enabled a more effective processing of DoD user requests, and allowed more time to be devoted to improving products and services.

More than 1700 documents have been distributed in response to the 900 document requests. Below is a list of the most requested documents.

Document Title	Number Distributed
DACS Glossary	734
SRR-1	469
BIB-1	345
Software Reliability Data	313
Software Data Collection and Analysis	270
Productivity Data Collection Forms	326

The figures more accurately reflect the results of the DACS user survey, than the figures reported for Phase II, in that the most requested documents were the state-of-the-art reports "The DACS Glossary" and SRR-1. Respondees of the DACS user survey indicated a preference for state-of-the-art reports.

12.2.3 Information Processing

As reported in Section 12.2.2.2, during Phase II we had concentrated on establishing a technology information base while continuing to serve the user. During Phase III we added to this information base, compilations and descriptions of software engineering research projects. An analysis was completed in February 1980 to determine the staff time needed for, and the

number of documents backlogged at each step of the processing of software engineering documents for this base. The results of this analysis are summarized below.

Step	Best Time	Worst Time	Avg Time	No. Backlogged at this Step
Log-in	5 min	26 min	16 min	416 documents
Code	5 min	15 min	8 min	220 documents
Index	5 min	20 min	10 min	322 documents
Verify	2 min	20 min	3 min	36 documents
Keypunch	4 days	1½ weeks	1 week	none
Load	3 hours	1 week	5 hours	none

The backlog of documents queued at each of the first three steps of the document processing path increased each month. For this reason, the DACS had not initiated an aggressive document acquisition program - we did not have the staff to process what was on hand an increase in acquisitions without a corresponding increase in staff would only increase the backlog.

The areas needing the most improvement were the log-in and coding steps. It was determined that the effort expended in these two steps could be reduced by removing some of the redundancy between these steps. After investigating the indexing step, we decided that little or no improvement could be made. It was, however, determined that the keypunching step could be improved. This improvement was based upon the expected continuing increase in the amount of relevant software engineering literature. With this increase it will be more cost effective to perform the keypunching in-house.

SECTION XIII

OBSERVATIONS AND RECOMMENDATIONS

13.1 Observations

Several conclusions may be made as a result of the experiences encountered during the first three years of operation of the DACS.

- Information on software engineering is generally available - but is widely scattered in many forms. For this reason it is difficult for software project managers, programmers, and other personnel who would find this information valuable, to gather and synthesize this information for practical use.

- Vast amounts of data are needed to support the broad spectrum of the software technology research needed to improve the quality, reliability and cost of computer software. Software development data is available on many software projects, however, data that is adequate for research is very limited. Most available datasets have been collected for project management rather than research purposes. As a result, they are often incomplete, inconsistent, incompatible or not applicable to many research programs. The problem is further compounded by the diversity of program applications, development environments, data elements collected, definitions used to define data elements, and research requirements. Aggressive data acquisition efforts are needed to obtain useful existing datasets and to promote and coordinate data collection activities on new projects.

- There are software tools available that provide powerful aids in designing, developing, testing, and maintaining computer software. However, it is difficult to integrate these tools into the development environment because of a lack of succint methodologies and information on evaluating the tools, providing justification for their use, and training programmers.

- A solid, usable base of software engineering information and data has been established and developed at the DACS.

- A framework for processing, synthesizing, analyzing, and disseminating this information and data has been established and developed at the DACS. This framework has been proven to be both, in keeping with the functions of an IAC and dynamic, in that it may be easily changed and enhanced as the functions of the DACS change.

- The products and services of the DACS are desired and needed by a wide range of users involved in software development and maintenance.

Generally, it may be concluded that the DACS has provided a unique

readily available, central source of information (outside of those existing and dedicated to individual organizations and companies) on all aspects of software engineering. Also, the framework for the processing, synthesis, analysis, and dissemination of this information has been proven to be viable. Furthermore, it has been demonstrated that the results of these activities are valuable to the users of the DACS.

13.2 Recommendations

Based on the observations drawn from the first three years of operation of the DACS, the following recommendations are proposed.

- Continue to expand the information base compiled during the previous three years of operation of the DACS. This is recommended because a current, readily available information base is necessary to maintain the high quality of services offered by the DACS. Additionally, users of the DACS are interested in the most recent information available and information becomes obsolete relatively quickly as research progresses.

- Aggressively promote the value of software experience data collection and actively pursue sources of software development and maintenance data. Since the DACS itself does not have the resources to collect data and software developers are often reluctant to collect data because of the added costs involved, it is necessary to convince developers that data is worth the effort to collect. Also, due to the sensitive nature of some of this data, it is also necessary to assure developers that data which they have collected will not compromise their organization if it is made available to the DACS. The DACS is in a unique position to objectively study software life cycle data from a variety of sources and this situation should be exploited.

- Continue to provide analyses of the DACS information and data and package the results of analysis into usable products for general distribution. As the amount of available data and information at the DACS increases, it will become increasingly important to analyze and disseminate this information on a timely basis, and in a form readily usable by the software engineering community.

- Continue to promote the products and services of the DACS through a comprehensive, unified program to maximize their contribution to software technology.

- Continue to solicite user feedback on the products and services offered by the DACS, through the methods developed during the pilot period to ensure that they will continue to provide the greatest usefulness to DACS users.

- Establish a cost recovery program which will allow generation of the additional resources necessary to further develop the DACS databases and its products and services.

Finally we recommend that the operation of the DACS be continued and expanded to further the field of software engineering, eventually contributing to the development of less expensive, higher quality, and more readily maintainable software.

REFERENCES

[BAKER-77] Baker, W. F., Software Data Collection and Analysis: A Real-Time System Project History, Rome Air Development Center, RADC-TR-77-192, Griffiss Air Force Base, NY, June 1977.

[BASILI-79] Basili, Victor R., Zelkowitz, Marvin V.; McGarry, Frank E.; Reiter, Robert W.; Truszkowski, Walter F.; Weiss, David L.; The Software Engineering Laboratory, TR-535, SEL-1, NASA Goddard Space Flight Center, Greenbelt, MD, May 1977.

[BISHOP-77] Bishop, Ethelyn and Nisenoff, Norman, An Application of Market Research Techniques to the Dissemenation of Scientific and Technical Information, National Science Foundation Report No. DS175-13211-A01, November 1977, 237 p.

[CRUM-77] Crum, F. B., et al., "Establishment and operation of Pilot Warranty Information Center," Arinc Research Corporation, "May 1977.

[CSC-79] Computer Sciences Corporation, A Review of Software Maintenance Technology, Contract F30602-78-C-0255, Rome Air Development Center, Griffiss Air Force Base, NY, November 1979.

[DUVALL-76] Duvall, Lorraine M., Software Data Repository Study, Rome Air Development Center, RADC-TR-76-387, Griffiss Air Force Base, NY, December 1976.

[DUVALL-79] Duvall, L. M., "A Software Reliability Review;" IIT Research Institute; March 1979.

[DUV-79I] Duvall, Lorraine M., Baseline Software Data System, System Description, Vol. 1 (of two), Rome Air Development Center, RADC-TR-79-185, Griffiss Air Force Base, NY, July 1979.

[DUV-79II] Duvall, Lorraine M., and Christine Curtis, Baseline Software Data System, Database Reference Manual, Vol. II (of two), Rome Air Development Center, RADC-TR-79-185, Griffiss Air Force Base, NY, July 1979.

[DUVALL-80] Duvall, L.M., Martens, J. B., Swearinger, D. and Donahoo, J., "Data Needs for Softwear Reliability Modeling"; 1980 Proceedings Annual Reliability and Maintainability Symposium.

[EHREN-78] Ehrenfried, C. E., Operation of the Reliability Analysis Center, Final Report, IIT Research Institute, November 1978.

[FRIES-77] Fries, M. J., Software Error Data Acquisition, Rome Air Development Center, RADC-TR-77-188, Griffiss Air Force Base, NY, June 1977.

[GACIAC] Conversations with personnel of Guidance and Control Information Analysis Center, a DoD IAC operated by IIT Research Institute, 10 West 35th St., Chicago, IL 60616.

[GAO-73] U. S. General Accounting Office. "Coordination of Computerized Information Systems. Reporting of Active Research Efforts."; Washington, DC: U. S. General Accounting Office (March 29, 1973) 12 p.

[GE-MTIAC] General Electric Company - Tempo, Planning Study to Establish DoD Manufacturing Technology Information Analysis Center, Report No. GE80 TMP-43, 1980, 193 p.

[GLOS-80] Gloss-Soler, S. and Duvall, L. M.; "Data Needs for Software Engineering",IIT Research Institute; October 1980.

[HECT-77] Hecht, H. et al., Reliability Measurement During Software Development, NASA Langley Research Center, NASA-CR-145205, Hampton, VA, September 1977.

[HERSEY-63] Hersey, David F. & Monroe E. Freeman. "User Responses in the Evaluation of a Flexible Indexing and Retrieval System" Proceedings of the 29th Annual Meeting of the American Documentation Institute, October 1963, Chicago IL pp. 117-118.

[KING-NCIC] King, D. W. and Roderer, N. K., A Study of Pricing Policies for Information Products and Services, prepared for U.S. Geological Survey, National Cartographic Information Center, March 1978, p. 146.

[MASON-77] Mason, Robert M., et al, Development of Cost Benefit Methodology for Scientific and Technical Information Communication and Application to Information Analysis Center, National Science Foundation Report No. SIS 75-12741, February 1977, p. 242.

[MEYER-79] Meyer, John D., "Analysis of the Roles and Benefits of a DoD Manufacturing Technology Information Center (MTIC), (Draft Report)," March 1, 1979.

[MUSA-79] Musa, John D., "Validity of the Execution Time Thoery of Software Reliability," IEEE Transactions on Reliability, August 1979.

[NELSON-78] Nelson, Richard, Software Data Collection and Analysis, Rome Air Development Center, Griffiss Air Force Base, NY, September 1978.

[RAC] Conversations with personnel of Reliability Analysis Center a DoD IAC operated by IIT Research Institute at Rome Air Development Center, Griffiss Air Force Base, NY 13440.

[RUBIN-77] Rubin, Stephen A., "Service Charging: An IAC Experience," printed in ASIS Sixth Mid-Year Meeting (The Value of Information), Suracuse University, Syracuse, NY, 1977.

[RYE-77] Rye, P., et al., Software Systems Development: A CSDL Project History, Rome Air Development Center, RADC-TR-77-213, Griffiss Air Force Base, NY, June 1977.

[SRR1-79] Computer Sciences Corporation, Quantitative Software Models SRR-1; Data and Analysis Center for Software; Rome Air Development Center; March 1979.

[THAY-76] Thayer, T. A., Software Reliability Study, Rome Air Development Center, RADC-TR-76-238, Griffiss Air Force Base, NY, August 1976.

[TRUN-81A] Turner, C. and G. Caron; DACS Data Compendium Series: NASA/SEL Software Development Data; Data and Analysis Center for Software; Rome Air Development Center; April 1981.

[TURN-81B] Turner, C. and G. Caron; DACS Technical Monograph Series: A Comparison of RADC and NASA/SEL Software Development Data; Data and Analysis Center for Software; Rome Air Development Center; May 1981.

[WILLIAM-77] William, H. E., Jr., et al., Software Systems Reliability: A Raytheon Project History, Rome Air Development Center, RADC-TR-77-188, Griffiss Air Force Base, NY, June 1977.

[WIT-80] Witter, Susan Z. Systems Analysis: Cost Recovery, Data & Analysis Center for Software, September 1980, 53 p.

[ZAIS-76] Zais, Harriet W., "Economic Modeling: An Aid to the Pricing of Information Services, Berkeley, CA: U.S. Energy Research and Development Administration, 1976. (NTIS No. CONF-761011-3).

APPENDIX A

PARTIAL LIST OF DACS USERS

APPENDIX A

PARTIAL LIST OF DACS USERS

United States Corporations & Individuals

Accutest Corporation
Chelmsford, MA

Advanced Information & Decision Systems
Mountain View, CA

Advanced Systems Incoterm Corp. *
San Diego, CA

Applied Research, Inc.
Santa Clara, CA

The Aerospace Corporation
Los Angeles, CA

American Airlines
Tulsa, OK

American Society of Civil Engineers
Kensington, MD

Analytic Sciences Corporation
Reading, MA

AMOCO Production Company
Tulsa, OK

Analytics
McLean, VA

Aquidneck Data Corporation
Middletown, RI

Argonne National Laboratory *
Argone, IL

Arthur Andersen & Company *
Chicago, IL

Arthur Young & Company
Los Angeles, CA

Automation Industry, Incorporated
Silver Spring, MD

Bell Laboratories *
(several locations)

Bell Northern Research *
Palo Alto, CA

Bendix Research Laboratories *
Southfield, MI

Bibliographic Services
Lexington, MA

Boeing Aerospace *
(several locations)

Booz, Allen, Hamilton
Bethesda, MD

Burroughs Corporation *
(several locations)

CBS Publishing Group
New York, NY

CTEC Corporation *
Falls Church, VA

Calculon Corporation *
Falls Church, VA

Carver, Doris L.
Baton Rouge, LA, CA

Control Data Corporation *
(several locations)

Cinncinnati Electronics
Cinncinnati, OH

Computer Sciences Corporation *
(several locations)

* Multiple Technical Inquiries

Computer World
Newton, MA

Connecticut General Life Insurance Co.
Hartford, CT

Consolidated Freightways *
Portland, OR

Cortex, Incorporated
Welfesley, MA

Cutler-Hammer
Long Island, NY

Data Systems Technical Informations Ctr.
New Haven, CT

John Deere Company *
Waterloo, IA

Digital Equipment Corporation *
Maynard, MA

Distributed Systems Corporation
Chelmsford, MA

Doty Associates, Incorporated *
Rockville, MD

Draper Laboratory *
Cambridge, MA

Douglas Aircraft Company *
Long Beach, CA

Dynamics Research Corporation *
Wilmington, MA

Eaton Corporation
Deer Park, NY

E-Systems, Incorporated
Falls Church, VA

EG & G, Idaho
Idaho Falls, ID

Electronics Associates, Incorporated
West Long Branch, NJ

Ernst Whinney
Cleveland, OH

ESL Corporation
Sunnyvale, CA

Evaluation Associates
Bala Cynwyd, PA

Environmental Research Institute of Michigan
Ann Arbor, MI

Ferguson-Bryan & Associates, Inc.
Washington, DC

Fireman's Fund Insurance Company *
San Rafael, CA

First National Bank of Atlanta
Atlanta, GA

Fisher Price Toys
East Aurora, NY

Ford Aerospace Company
Sunnyvale, CA

Ford Motor Company
Dearborn, MI

Gagliardi Systems Group (GSG) *
Rome, NY

General Dynamics *
(several locations)

General Electric Company *
(several locations)

General Research Corporation *
(several locations)

General Telephone & Electronics
Needham Heights, MA

Goodyear Aerospace
Akron, OH

Grumman Aerospace Corporation *
Bethpage, NY

* Multiple Technical Inquiries

Harris Corporation *
Melbourne, FL

Health Products Research, Inc.
Somerville, NJ

Higgins, James S.
Houston, TX

Higher Order Software
Jericho, NY

Hitachi American Ltd.
Atlanta, GA

Honeywell, Incorporated *
(several locations)

Hughes Aircraft Company *
(several locations)

IEEE Computer Society
Huntington Beach, CA

IIT Research Institute *
(several locations)

Inco, Incorporated
McLean, VA

Input Corporation
Saddlebrook, NJ

International Business Machines *
(several locations)

ITT Telecommunications Technical Ctr.
Stratford, CT

Jet Propulsion Laboratory
Pasadena, CA

Linkabit Corp.
San Diego, CA

Litton-Mellonics
Springfield, VA

Lockheed Electronics *
Plainfield, NJ

Lockheed Georgia Company
Marietta, GA

Lockheed Missiles & Space Co., Inc. *
Sunnyvale, CA

Logicon, Incorporated *
San Pedro, CA

Manufacturing Data Systems, Inc.
Ann Arbor, MI

Martin-Marietta Corporation *
(several locations)

McCabe & Associates *
Columbia, MD

McDonald Douglas Auto Company
Long Beach, CA

McDonnell Douglas Corporation *
(several locations)

Arthur G. McKee & Company
Cleveland, OH

Measurement Concept Corporation *
Rome, NY

Memorex
Santa Clara, CA

Mitre Corporation *
Bedford, MA

NCR Corporation *
San Diego, CA

Norden Systems
Nowalk, CT

Northeast Utilities *
Hartford, CT

Northrop Electronics Division
Hawthorne, CA

Oak Communications, Incorporated
Rancho Bernardo, CA

* Multiple Technical Inquiries

OAO Corporation *
El Segundo, CA

Peat, Marwick, Mitchell & Company *
New York, NY

Pensions Benefit Guaranted Corporation
Silver Springs, MD

Perkin-Elmer *
Danbury, CT

Planning Research Corporation *
(several locations)

Plateau Oil Company
Belen, NM

Pressman & Hartunian, Ltd.
Chicago, IL

Price Waterhouse & Company *
New York, NY

Quantitative Software Management, Inc. *
McLean, VA

Pardyne, Corporation
Largo, CA

The Rand Corporation *
Santa Monica, CA

Raytheon Company *
(several locations)

Raytheon Service Company
Arlington, VA

Reliance Corporation
Cleveland, OH

Research Triangle Institute
Research Triangle Park, NC

Rockwell International *
Cedar Rapids, IA

RCA Government Communications System
Camden, NJ

RTA Associates
Lockport, IL

SAI Comsystems Corporations *
San Diego, CA

Sander Associates
Nashua, NH

Science Applications, Inc. *
Englewood, CO

Sheridan Oaks Cybernetics
Glenview, IL

Singer, Link Division
Binghamton, NY

Software Enterprises Corporation *
Westlake Village, CA

Softech Incorporated *
Waltham, MA

Software Management Consultants *
Torrance, CA

Software Research Associates
San Francisco, CA

SoHar Incorporated *
Los Angeles, CA

Southern New England Telephone Co. *
New Haven, CT

Southwest Research Institute
San Antonio, TX

Spectra Medical Systems *
Sunnyvale, CA

Sperry Rand Corporation
(several locations)

Sterling Systems
McLean, VA

Stone, Harold
Amherst, MA

* Multiple Technical Inquiries

Sun Information Services
Philadelphia, PA

Support Systems Associates, Inc.
Coronado, CA

Surgical Associates
Albuquerque, NM

Syntex Corporation
Palo Alto, CA

System Development Corporation *
Santa Monica, CA

Systems and Applied Sciences Corp. *
Riverdale, MD

Systems Engineering Laboratories *
Fort Lauderdale, FL

Sylvania (GTE), Incorporated *
Needham, MA

Tektronix, Incorporated
Beaverton, OR

Teletype Corporation *
Chicago, IL

Tera Industrial Controls, Inc.
Austin, TX

Texas Highway Department
Austin, TX

Texas Instruments *
(several locations)

Time Share Corporation
Hanover, NH

Triad Microsystems, Incorporated *
Huntsville, AL

Union Carbide Company *
(several locations)

TRW *
Redondo Beach, CA

* Multiple Technical Inquiries

Union Gas Ltd.

Veda, Incorporated
Dayton, OH

Western Electric Company *
New York, NY

Westinghouse ILSD Division *
Hunt Valley

Xerox Corporation *
Rochester, NY

Youngdahl, Coleman G.
Sonoma, CA

Zimmerman & Associates
Rockville, MD

Department of Defense

Department of Defense
Ft. Meade, MD

Defense Documentation Center
Alexandria, VA

Defense Logistics Agency *
Alexandria, VA

Defense Mapping Agency *
Washington, DC

Defense System Acquisition Branch
ADP Acquisition Improvement Project
Washington, DC

Office of the Assistant Secretary
of Defense
Washington, DC

TRI-TAC
Ft. Monmouth, NJ

U.S. Air Force

AF/ACDR
Fairfax, VA

Air Force Data Services Center *
AFDSC/XMD Pentagon
Washington, DC

Air Force Reserve *
SD/CSB RES. DET 1
Santa Clara, CA

Air Force Station/USAF
Requirements & System Engineering
Liaison
Sunnyvale, CA

Andrews Air Force Base *
AFSC/XRF
Washington, DC

Edwards Air Force Base
Air Force Flight Task Center
Edwards, CA

Griffiss Air Force Base *
RADC
Rome, NY

Gunter Air Force Base *
Phase IV PMO/PGY
and
AF DSDS/SCDP
Montgomery, AL

Hanscom Air Force Base *
AFTL Laboratory
and
AFGL/SULLR
and
ESD/TOI, TOIS, TOIT, TOIQ
TOIT Hqs. ESD
Bedford, MA

HQ USAF-ACDB
Washington, DC

Kirtland Air Force Base *
HQS AFCDM/QA
and
Air Force Testing Evaluation Center
NM

Langley Air Force Base
HQS TAC/ADYS
VA

McClellan Air Force Base
Sacramento Air Logistics Center
Air Force Logistics Command
and
SM-ALC/MMM-1
and
ACD
CA

Maxwell Air Force Base
ACSC/EDOI
AL

Nellis Air Force Base *
Tactical Fighter Weapons Center
SA/M
NV

Offutt Air Force Base *
HQ SAC
ADOS
and
HQS SAC/ADXPO
Global Weather Central/DOY
Global Weather Central/ADSV
NB

Plattsburg Air Force Base
4200 Test Evaluation Squadron
Det 3
NY

Randolph Air Force Base
AFMPG/MPCD
TY

Robbins Air Force Base *
WR/ALC/MMEDV
GA

* Multiple Technical Inquiries

Scott Air Force Base *
HQ MAC/ADLTC
IL

Tinker Air Force Base *
AFCCRC
OK

USAF/AFPRO
Sunnyvale, CA

Wright-Patterson Air Force Base *
AFLC
and
AFAL/AAA-3
AFFA/ASD
ASD/YPAA
AFLC/ACMCE
OH

U.S. Army

Department of the Army
Automated Logistic Management
Systems Agency
St. Louis, MO

U.S. Army
Ft. Huachuca, AZ

U.S. Army *
AIRMICS
Atlanta, GA

U.S. Army
Defense Management College
Ft. Belvoir, VA

U.S. Army
ERADCOM
Ft. Monmouth, NJ

U.S. Army
HQ Seventh Transportation Group
Ft. Eustis, VA

U.S. Army Research Institute for
the Behavioral and Social Sciences
Alexandria, VA

* Multiple Technical Inquiries

U.S. Navy

Commander Frank Boebert, USN
DCASR
New York, NY

David Taylor Naval Ship
R & D Center
Bethesda, MD

Naval Air Development Center
Warminster, PA

Naval Air Systems Command
Reliability/Maintainability
Washington, DC

Naval Aviation Logistic Center *
Naval Air Station
Patuxent River, MD

Naval Ocean Systems Center *
San Diego, CA

Naval Research Code #474
Arlington, VA

Naval Research Laboratory *
Code 7906 & 7503
Washington, DC

Naval Sea Systems Command *
Code 63 E 11
Washington, DC

Naval Surface Weapons Center *
Code K70
and
Dahlgren Laboratory
Dahlgren, VA

Naval Systems Weapon Command *
Code K74
Dahlgren, VA

Naval Training Equipment Center *
Code N-4133
Orlando, FL

Naval Underwater Systems Center *
Codes 3545, 3552, 411, 715
Newport, RI

Naval Weapons Center
China Lake, CA

Norfolk Naval Base
CINCLANT
Norfolk, VA

Other Government Agencies

Department of Health, Education,
and Welfare
Public Health Service
Bethesda, MD

Department of Water and Power
City of Los Angeles
CA

Elementary Particle Information Ctr.
Oak Ridge National Laboratory
Oak Ridge, TN

Federal Aviation Administration
Margate, NJ

FEDSIM/NA
Washington, DC

General Services Administration (GSA) *
Washington, DC

Institute for Defense Analysis
Arlington, VA

NASA-AMES Research Center *
Moffett Field
CA

NASA, Goddard Space Flight Center *
Greenbelt, MD

NASA, Langley Research Center
Hampton, VA

National Atmospheric and Oceanic
Administration
Washington, DC

National Bureau of Standards (NBS)*
Washington, DC

National Security Agency (NSA) *
Ft. George G. Meade
MD

Nuclear Regulatory Commission
Washington, DC

USA/OTEA
Operations Test Evaluation Agency
Falls Church, VA

U.S. Department of State
Fairfax, VA

U.S. Department of Transportation
Office of Automated Systems Policy
Washington, DC

U.S. Environmental Protection
Agency
Research Triangle Park
NC

U.S. General Accounting Office *
(several locations)

U.S. Social Security Administration
(several locations)

Veterans Administration
Washington, DC

Universities

Arizona State University
Tempe, AZ

California State University at
Northridge
Northridge, CA

Carnegie-Mellon University
Pittsburgh, PA

Case Western Reserve University
Cleveland, OH

* Multiple Technical Inquiries

Colorado State University *
Ft. Collins, CO

Davis and Elkins College
Elkins, WV

Florida Atlantic University *
Boca Raton, FL

Georgia Institute of Technology *
Atlanta, GA

Iowa State University
Ames, IA

New Mexico Institute of Technology
Socorro, NM

Ohio State University
Columbus, OH

State University of New York
at Binghamton
Binghamton, NY

Syracuse University
Syracuse, NY

The Johns Hopkins University
Laurel, MD

University of Colorado
Colorado Springs, CO

University of Delaware
Newark, DE

University of Maryland
College Park, MD

University of Minnesota, MISRC *
Minneapolis, MN

University of Missouri
Rolla, MO

University of New Mexico
Albuquerque, NM

University of Rochester *
Rochester, NY

University of Southern California *
Los Angeles, CA

University of South Florida
Tampa, FL

* Multiple Technical Inquiries

APPENDIX B

SUBJECTS OF TECHNICAL INQUIRIES AND BIBLIOGRAPHIC REQUESTS

APPENDIX B

SUBJECTS OF TECHNICAL INQUIRIES AND BIBLIOGRAPHY REQUESTS

This appendix presents a summary of the topics of interest to DACS users as measured through the subjects of technical inquiries and bibliographic searches performed by the DACS.

The Service Request Record (SVC) illustrated in Figure B-1 has been used as a vehicle to track and record services performed by the DACS in terms of the user (requestor), the type of service provided, and the resources expended. The information has been transferred to the User Profile Record (UPR) to accumulate the types of services provided each user (Figure B-2). Periodic tabulations made of total user interactions with the DACS and correlated with the results of periodic user surveys to evaluate the Center's responsiveness and used as input to the further development and enhancement of DACS products and services.

During the 35-month period ending August 15, 1981, the DACS received 832 technical inquiries from individuals associated with more than 250 organizations.

While some of the inquiries could be satisfied by material contained in a DACS publication, i.e., SRR-1; others by referral to another agency, e.g., inquiries about obtaining government developed software, were referred to the Federal Software Exchange Center; the vast majority required research by DACS personnel or bibliographic searches or both. Seven hundred and six of the inquiries fell into this group. The subjects of those seven hundred and six inquiries were tabulated to provide a picture of the areas of greatest concern to DACS users.

DACS SERVICE REQUEST RECORD

SVC#

NAME AND TITLE

DATE | TIME | MODE

SEVERITY
- ☐ ROUTINE
- ☐ BY _____
- ☐ ASAP

AGENCY/COMPANY AND DIVISION

APPLICATION
- ☐ GOVT ☐ COMMERCIAL
- ☐ CONTRACT ☐ PROPOSAL

ADDRESS-INCLUDE BUILDING AND MAIL STOP

AGENCY

UPR #

CITY | STATE | ZIP

APPLICATION AREA

REQUEST CATEGORY
- ☐ CONSULTING ☐ DATABASE
- ☐ INFORMATION ☐ BIBLIOGRAPHY
- ☐ OTHER

PHONE: AREA CODE | NUMBER | EXTENSION

PROBLEM STATEMENT (ATTACH ADDITIONAL SHEETS IF NECESSARY)

RECOMMENDED ACTION (TO BE COMPLETED BY CENTER PERSONNEL)

AUTHORIZED SIGNATURE _____ DATA ANSWERED _____

RESPONSE (TO BE COMPLETED BY CENTER PERSONNEL)

LIAISON | HRS

EST. VALUE OF SVCS

ANALYST | HRS

DATE COMPLETED | MODE

FIGURE B-1

DACS SERVICE REQUEST RECORD

DACS USER PROFILE RECORD

UPR#

AGENCY/COMPANY AND DIVISION			DATE	SUBSCRIBER TYPE
ADDRESS-INCLUDE BUILDING AND MAIL STOP			TECHNICAL CONTACTS	
CITY	STATE	ZIP	ADMINISTRATIVE CONTACTS	
PHONE: AREA CODE	NUMBER	EXTENSION		
COMPUTER SYSTEM-NAME	MODEL OR TYPE		SPECIAL FACTORS	

DOCUMENT NAME	VERSION	DATE SENT	MAJOR APPLICATIONS

ATTACH ADDITIONAL SHEETS IF NECESSARY

DATASET NAME	USAGE	DATE SENT OR RECEIVED	MAJOR APPLICATIONS

SVC#			

FIGURE B-2

DACS USER PROFILE RECORD

The areas of highest concern are depicted in Figure B-3. In addition, another 103 inquiries, or 14.6% of the total, were made with regard to 43 other topics. Figure B-4 displays the number of inquiries processed in closely related topics.

Figure B-5 displays the types of organizations which make technical inquiries to the DACS. These figures were obtained from UPRs.

Technical Inquiry Topic	Number of Requests	Percentages
Costs	60	8.5%
Quality Assurance	34	4.8%
Testing	32	4.5%
Software Tools	29	4.1%
Quality Factors/Metrics	28	4.0%
Reliability	28	4.0%
Management	26	3.7%
Verification & Validation	25	3.5%
Conversions	24	3.4%
Maintenance	23	3.2%
Development	20	2.8%
Modelling/Simulation Tools	20	2.8%
Languages	20	2.8%
Standards/Standardization	20	2.8%
Design	17	2.4%
Errors/Faults	17	2.4%
Documentation	16	2.3%
Datab › Management Systems	15	2.1%
Produ ivity	15	2.1%
Configuration Management	14	2.0%
Requirements	14	2.0%
Data/Data Collection	13	1.8%
Modern Programming Practices	10	1.4%
Dumps	9	1.3%
Distributed Processing	9	1.3%
Operating System Design	8	1.1%
Embedded Computers	7	1.0%
DoD Common HOL (ADA)	7	1.0%
Acquisition Management	7	1.0%
Failures	6	0.8%
Real Time Applications	6	0.8%
Security	6	0.8%
Database Applications	6	0.8%
Structured Programming	6	0.8%
Performance Measurement	6	0.8%

FIGURE B-3

TOPICS OF TECHNICAL INQUIRIES

Combined Groupings of Topics	Number of Requests	Percentages
Costs and Productivity	75	10.6%
Quality Factors/Metrics; Reliability; Complexity	60	8.5%
Validation & Verification; Testing; Acceptance Testing	58	8.2%
Software Tools; Modelling & Simulation Tools	49	6.9%
Quality Assurance	34	4.8%

FIGURE B-4

CLOSELY RELATED TOPICS OF INTEREST

Organization Catagory	Number of Organizations	Percentage
U.S. Air Force	22	8.2%
U.S. Navy	14	5.2%
U.S. Army	8	3.0%
TOTAL DoD	44	16.4%
Other Government Agencies	30	11.2%
TOTAL GOVERNMENT	74	27.5%
Universities/Academia	27	10.0%
Industry	168	62.5%
TOTAL	269	100.0%

FIGURE B-5

USER PROFILE STATISTICS

APPENDIX C

STATISTICS ON KEYWORD DISTRIBUTION

IN THE SEB DATABASE

APPENDIX C

STATISTICS ON KEYWORD DISTRIBUTION

This Appendix contains an alphabetical listing of the 519 keywords in the DACS Thesaurus together with the number of documents assigned each keyword. There were 12,767 keyword assignments made to 2,158 online document descriptions, yielding an average of six keywords per document.

The keywords assigned most often are:

Keyword	Number of Instances
SOFTWARE TOOLS	212
STRUCTURED PROGRAMMING	181
DEVELOPMENTAL TOOLS AND TECHNIQUES	166
MANAGEMENT TOOLS AND TECHNIQUES	152
RELIABILITY	148
PROGRAMMING LANGUAGE	128
DEVELOPMENT MANAGEMENT	126
AUTOMATED VERIFICATION TOOLS	125
PROGRAMMING TECHNIQUES/METHODOLOGIES	120
DOCUMENTATION	117
HIGHER-ORDER LANGUAGES	116
TESTING	115
SPECIFICATIONS	106
SYSTEM DESIGN	104
FAULT DETECTION	104
DISTRIBUTED PROCESSING	103
CONFIGURATION MANAGEMENT	102
MODELLING AND SIMULATION TOOLS	101
AUTOMATED TESTING	100
VERIFICATION	99
LANGUAGE EVALUATION	97
DEVELOPMENTAL PROCESS	97
RELIABILITY MODELS	96
DATA STRUCTURES	96
DESIGN TOOLS AND TECHNIQUES	95
SOFTWARE ENGINEERING TOOLS AND TECHNIQUES	94
DEVELOPMENTAL METHODOLOGIES	94
DESIGN METHODOLOGIES	94
SOFTWARE ENGINEERING	92
COST AND SCHEDULE CONTROL	91
COMPILERS	91
QUALITY ATTRIBUTES	90
TEST DATA GENERATION	88
STANDARDS	88
PROGRAM CORRECTNESS	86
PROGRAM TESTING	85
REQUIREMENTS	84
LIFE CYCLE COSTS	83

KEYWORD	NO. DOCUMENTS INDEXED USING THIS KEYWORD
ABSTRACT MACHINES	12
ACCEPTANCE TESTING	12
ACCESSIBILITY	9
ACCESS-CONTROL MECHANISMS	28
ACQUISITION COSTS	6
ACQUISITION MANAGEMENT	77
ADAPTABILITY	5
ALGOL	18
ALGORITHM ANALYSIS	28
ALPHARD	5
APL (A PROGRAMMING LANGUAGE)	6
APPLICATION-ORIENTED LANGUAGES	28
APPLICATIONS	20
ARCHITECTURAL FAMILIES	11
ARCHITECTURE	24
ARTIFICIAL INTELLIGENCE	6
ASSEMBLERS	23
ASSEMBLY LANGUAGE	8
ASSERTIONS	30
ASSIGNMENT STATEMENTS	2
AUDITS	10
AUGMENTABILITY	1
AUT (AUTOMATED UNIT TEST)	3
AUTOMATA	3
AUTOMATA THEORETIC	1
AUTOMATED DESIGN TOOLS	22
AUTOMATED DOCUMENTATION	36
AUTOMATED FAULT DETECTION	58
AUTOMATED PROGRAM ANALYSIS	28
AUTOMATED TESTING	100
AUTOMATED VERIFICATION TOOLS	125
AUTOMATIC DATA COLLECTION	50
AUTOMATIC PROGRAMMING	34
AUTOMATIC SOFTWARE TEST DRIVERS	8

KEYWORD	NO. DOCUMENTS INDEXED USING THIS KEYWORD
AVAILABILITY	14
AVAILABILITY MODELS	6
AVIONICS APPLICATIONS	69
BACKEND DATABASES	1
BALG	1
BALLISTIC MISSILE DEFENSE	20
BASIC	12
BEHAVIOR MODELLING	7
BOTTOM UP DESIGN	6
BUG SEEDING/TAGGING	15
BUILDS	3
BUSINESS AND FINANCIAL APPLICATIONS	11
C LANGUAGE	2
CAPABILITY MACHINES	5
CERTIFICATION	9
CHIEF PROGRAMMER TEAM	26
CLARITY	12
CLU	2
COBOL	35
CODE READING	11
CODE VERIFICATION	14
COMMAND LANGUAGES	7
COMMUNICATIONS SWITCHING SYSTEMS	28
COMMUNICATIVENESS	19
COMPILER-COMPILERS	21
COMPILERS	91
COMPLEXITY	51
COMPLEXITY MEASUREMENT	63
COMPUTATION STRUCTURES	5
COMPUTER COMMUNICATIONS NETWORKS	41
COMPUTER PROGRAM DEVELOPMENT PLAN (CPDP)	4
CONCISENESS	10
CONCURRENT PASCAL	17
CONCURRENT PROGRAMMING	34

KEYWORD	NO. DOCUMENTS INDEXED USING THIS KEYWORD
CONDITION VARIABLES	2
CONFIGURATION MANAGEMENT	102
CONFINEMENT	2
CONTROL STRUCTURES	56
CONVERSION AIDS	15
CONVERSION COST FACTORS	6
CONVERSION COSTS	10
CONVERSIONS	20
CORRECTNESS PROOFS	77
COST	39
COST AND SCHEDULE CONTROL	91
COST EFFECTIVENESS	41
COST ESTIMATION	76
COST FACTORS	23
COSTING TECHNIQUES	10
COST-BENEFIT ANALYSIS	59
COST/PRODUCTIVITY MODELS	23
CURRICULA	22
DAS (DESIGN ANALYSIS SYSTEM)	2
DATA	15
DATA ANALYSIS	82
DATA COLLECTION	80
DATA COLLECTION COSTS	6
DATA DOMAINS	30
DATA FLOWGRAPHS	34
DATA PARAMETERS	63
DATA PROCESSING APPLICATIONS	54
DATA REPOSITORY	12
DATA RESTRUCTURING	4
DATA SEMANTICS	14
DATA STRUCTURES	96
DATA VALIDATION	4
DATABASE MANAGEMENT SYSTEMS	81
DEBUGGING	79
DEBUGGING MODELS	8
DECISION TABLES	2
DESIGN	56
DESIGN ANALYSIS	43
DESIGN ANALYZERS	8
DESIGN LANGUAGES	12
DESIGN METHODOLOGIES	94
DESIGN REVIEWS	13
DESIGN TOOLS AND TECHNIQUES	95
DESIRED BEHAVIOR SPECIFICATION	4
DEVELOPMENT	28
DEVELOPMENT CYCLE	69
DEVELOPMENT MANAGEMENT	126
DEVELOPMENT SUPPORT LIBRARIAN	20
DEVELOPMENTAL METHODOLOGIES	94
DEVELOPMENTAL PROCESS	97
DEVELOPMENTAL TOOLS AND TECHNIQUES	166
DIFFICULTY	11
DIGITAL AIRCRAFT CONTROL	15
DISCRETE SYSTEMS ANALYSIS	1
DISTRIBUTED PROCESSING	103
DOCUMENTATION	117
DOD COMMON HIGH ORDER LANGUAGE	26
DOMAINS	3
DOTY MODEL	1
DREAM	4
DYNAMIC REDUNDANCY	9
EDITORS	5
EDUCATION	37
EFFICIENCY	51
EMBEDDED COMPUTER SYSTEMS	59
EMBEDDED LANGUAGES	37
EMULATION	23
EQUIVALENCE OF MONITOR IMPLEMENTATION	1

KEYWORD	NO. DOCUMENTS INDEXED USING THIS KEYWORD
ERROR ANALYSIS	61
ERROR CATEGORIES	1
ERROR MODELS	63
ERRORS	34
EUCLID	5
EVENT-BASED MODELS	2
EVOLUTIONARY SYSTEMS	5
EXECUTION TIME	3
EXECUTION TIME THEORY	16
EXPOSURE	1
EXTENSIBILITY	11
FAILURE CATEGORIES	2
FAILURE DATA	23
FAILURE RATES	22
FAILURE RATIO	25
FAILURES	18
FAIRNESS	2
FAST	1
FAULT AVOIDANCE	5
FAULT CATEGORIES	33
FAULT CATEGORIZATION	34
FAULT CONFINEMENT	8
FAULT CORRECTION	23
FAULT CORRECTION COSTS	20
FAULT CORRECTION LIMIT POLICIES	8
FAULT DATA	33
FAULT DATA ACQUISITION	9
FAULT DATA ANALYSIS	35
FAULT DETECTION	104
FAULT DETECTION CODE	2
FAULT DETECTION PROCESS	8
FAULT MANAGEMENT	9
FAULT PREVENTION	10
FAULT SEVERITY	4

KEYWORD	NO. DOCUMENTS INDEXED USING THIS KEYWORD
FAULT TOLERANCE	10
FAULT-TOLERANT SOFTWARE	50
FAULTS	9
FILE MANAGEMENT SYSTEMS	2
FILE MODIFICATION	2
FILE MODIFICATION PROTECTION	2
FIRMWARE	3
FLEXIBILITY	11
FORTRAN	72
FUNCTION TESTING	5
FUNCTIONAL PROGRAMMING	6
FUNCTIONS	5
GEOMETRIC-POISSON MODEL	1
GOEL-OKUMOTO MODEL	1
GYPSY	6
HAL/S	7
HALSTEAD'S LAW	13
HARDWARE RELIABILITY	6
HARDWARE/SOFTWARE TRADEOFFS	16
HAZARD FUNCTION	8
HEURISTIC SEARCH	4
HIERARCHIAL STRUCTURE	22
HIERARCHY	13
HIGHER-ORDER LANGUAGES	116
HIPO (HIERARCHY PLUS INPUT-PROCESS-OUTPUT)	10
HOLDET	1
HUMAN ENGINEERING	45
IBM (WALSTON AND FELIX) MODEL	1
IMPERFECT DEBUGGING	11
IMPLEMENTATION	73
IMPLEMENTATION CORRECTNESS	8
INDUCTIVE ASSERTIONS	13
INDUSTRIAL PROCESS APPLICATIONS	15
INFERENCE RULES	3

KEYWORD	NO. DOCUMENTS INDEXED USING THIS KEYWORD
INFORMATION HIDING	13
INFORMATION SYSTEMS	62
INPUT-DATA SENSITIVITY	3
INTEGRITY	5
INTERFACE CONTROL	3
INTERMITTENT ASSERTION METHOD	1
INTEROPERABILITY	3
INTERPRETATION CORRECTNESS	2
INTERPRETERS	9
INTERPROCESS COMMUNICATION	12
INVARIANT ASSERTIONS	17
ISRAD	2
JAVS	9
JELINSKI-MORANDA MODEL	18
JOINT LOGISTICS COMMANDERS	1
JOVIAL	30
KERNEL	2
KNOWLEDGE BASED SYSTEMS	5
LANGUAGE DESIGN	51
LANGUAGE EVALUATION	97
LANGUAGE STRUCTURE	8
LANGUAGES	26
LEGIBILITY	15
LEIGHTON DIAGRAMS	2
LIFE CYCLE COSTS	83
LINKAGE EDITORS	6
LIS	1
LISP	3
LIST PROCESSING	4
LOADERS	3
LOGISTICS APPLICATIONS	8
MACROPROCESSORS	12
MACRO-RELIABILITY MODELS	1
MAINTAINABILITY	59
MAINTENANCE	76
MAINTENANCE COSTS	33
MAINTENANCE TOOLS AND TECHNIQUES	64
MANAGEMENT	57
MANAGEMENT TOOLS AND TECHNIQUES	152
MATHEMATICAL METHODOLOGIES	67
MEDICAL APPLICATIONS	10
MEMORY MANAGEMENT	27
MESA	1
MESSAGE TRANSFER MODELS	1
METALANGUAGES	9
MICRO COMPUTERS	31
MICRO RELIABILITY MODELS	3
MICROCODE	3
MICROPROCESSORS	63
MICROPROGRAMS	19
MINICOMPUTERS	33
MODEL (PROGRAMMING LANGUAGE)	1
MODELLING AND SIMULATION TOOLS	101
MODELS	46
MODERN PROGRAMMING PRACTICES	66
MODIFIABILITY	16
MODIFICATION	22
MODIFICATION PROCEDURES	54
MODULA	4
MODULAR DECOMPOSITION	30
MODULAR PROGRAMMING	39
MODULARITY	59
MODULARIZATION	16
MODULE ANALYSIS	5
MODULE SIZING	2
MODULE TESTING	13
MODULES	7

KEYWORD	NO. DOCUMENTS INDEXED USING THIS KEYWORD
MONITORS	30
MULTICS	4
MULTIPROGRAMMING	21
MUSA'S MODEL	9
MUST	6
MUTUAL EXCLUSION	3
NATURAL LANGUAGE	6
NATURAL LANGUAGE PROCESSING	7
NATURAL LANGUAGE THEORY	5
NETWORKS	34
NON-PROCEDURAL SPECIFICATION	2
N-VERSION PROGRAMMING	2
NUCLEAR REACTOR APPLICATIONS	7
NUMERICAL MANIPULATION	1
OPAL	1
OPERATING SYSTEM DESIGN	27
OPERATING SYSTEMS	73
OPERATIONAL RELIABILITY	6
OPTIMIZATION	49
PANVALET	1
PARTITIONING	8
PASCAL	61
PATH EXPRESSIONS	15
PERFORMANCE	41
PERFORMANCE EVALUATION	78
PERSONNEL MANAGEMENT	8
PETRI NETS	8
PL/I	20
POINTERS	2
PORTABILITY	68
PREPROCESSORS	3
PRIVILEGE	11
PROBLEM REPORT ANALYSIS	5
PROCEDURES	12
PROCESS	2
PROCESS CONSTRUCTION	4
PROCESS DESIGN LANGUAGE (PDL)	4
PROCESS QUEUES	9
PRODUCT LIABILITY	2
PRODUCT SAFETY	1
PRODUCT SAFETY EVALUATION	1
PRODUCTIVITY	38
PRODUCTIVITY DATA	4
PRODUCTIVITY FACTORS	38
PROGRAM ANALYSIS	42
PROGRAM ANNOTATION	1
PROGRAM COMPLEXITY	25
PROGRAM CONTRACTION	1
PROGRAM CORRECTNESS	86
PROGRAM DESIGN	23
PROGRAM DESIGN LANGUAGE (PDL)	20
PROGRAM DESIGN METHODOLOGIES	29
PROGRAM EVALUATION	2
PROGRAM EXTENSION	4
PROGRAM FAMILIES	8
PROGRAM LIBRARY SYSTEMS	24
PROGRAM MAINTENANCE	26
PROGRAM PROTECTION	1
PROGRAM SYNTHESIS	19
PROGRAM TESTING	85
PROGRAM TRANSFORMATIONS	22
PROGRAM UNDERSTANDING	7
PROGRAM VALIDATION	13
PROGRAMMER PRODUCTIVITY	80
PROGRAMMER TRAINING	2
PROGRAMMER'S APPRENTICE	4
PROGRAMMING	35
PROGRAMMING AIDS	61

KEYWORD	NO. DOCUMENTS INDEXED USING THIS KEYWORD
PROGRAMMING LANGUAGE	128
PROGRAMMING LANGUAGE COMPLEXITY	9
PROGRAMMING TECHNIQUES/METHODOLOGIES	120
PROGRAMS	16
PROJECT MANAGEMENT SURVEYS	15
PROJECT WORKBOOK	2
PROTECTION	26
PROTECTION MEASUREMENT	2
PROTOCOLS	15
PUTNAM (SLIM) MODEL	1
QUALITY	15
QUALITY ASSURANCE	58
QUALITY ATTRIBUTES	90
QUALITY METRICS	56
QUEUING	14
RCA (PRICE-S) MODEL	14
REAL-TIME SYSTEMS	58
RECOVERY	26
RECOVERY BLOCK STRUCTURED SOFTWARE	14
RECURSION	7
REGRESSION TESTING	19
RELATIONAL DATA MODEL	2
RELIABILITY	148
RELIABILITY ESTIMATION	30
RELIABILITY EVALUATION	17
RELIABILITY MEASUREMENT	68
RELIABILITY MODELS	96
RELIABILITY PREDICTION	42
RELIABILITY TREND	15
RELIABILITY--DIFFERENCES OF OPINION	3
REQUIREMENTS	84
REQUIREMENTS ANALYSIS	60
REQUIREMENTS ENGINEERING	28
REQUIREMENTS ENGINEERING METHODOLOGIES	16
REQUIREMENTS LANGUAGES	26
REQUIREMENTS PROBLEMS	3
REQUIREMENTS SPECIFICATIONS	25
RESERVATIONS AND DISPATCHING APPLICATIONS	3
RESILIENCY	3
RESOURCE ALLOCATION	37
RESYNCHRONIZATION	3
REUSABILITY	18
ROBUSTNESS	6
ROLLBACK	7
ROUTING DATA	2
SAMM (SYSTEMATIC ACTIVITY MODELING METHOD)	4
SARA (SYSTEMS ARCHITECT'S APPRENTICE)	2
SCHEDULE ESTIMATION	29
SCHEDULING	20
SCHICK-WOLVERTON MODEL	14
SECURITY	32
SELF-DESCRIPTIVENESS	15
SEMANOL	9
SEMS	7
SETL	1
SHOOMAN MODEL	11
SIGNALLING	2
SIMULA	3
SIMULATORS	19
SIZING	15
SLICE	1
SNOBOL (AND SNOBOL EXTENSIONS)	7
SOFTWARE	13
SOFTWARE DATABASE	40
SOFTWARE ENGINEERING	92
SOFTWARE ENGINEERING BLUEPRINTS	3
SOFTWARE ENGINEERING FACILITY	23
SOFTWARE ENGINEERING PROJECT MANAGEMENT	44

KEYWORD	NO. DOCUMENTS INDEXED USING THIS KEYWORD
SOFTWARE ENGINEERING STANDARDS	13
SOFTWARE ENGINEERING TOOLS AND TECHNIQUES	94
SOFTWARE EXPERIENCE DATA	31
SOFTWARE LIBRARY MANAGEMENT SYSTEM	4
SOFTWARE LIFE CYCLE	48
SOFTWARE PHYSICS	6
SOFTWARE PRODUCT	5
SOFTWARE REQUIREMENTS ENGINEERING PROGRAM	3
SOFTWARE SCIENCE	22
SOFTWARE SNEAK ANALYSIS	1
SOFTWARE TERMINOLOGY	1
SOFTWARE TOOLS	212
SPARCS-II	2
SPECIFICATION LANGUAGES	24
SPECIFICATION TOOLS AND TECHNIQUES	49
SPECIFICATIONS	106
SPECIFICATIONS DECOMPOSITION	2
SPECIFICATIONS VERIFICATIONS	20
SPL (SYMBOL PROGRAMMING LANGUAGE)	1
STANDARDIZATION	43
STANDARDS	88
STATE DIAGRAMS	6
STATE MACHINES	20
STATIC ANALYSIS	26
STATIC REDUNDANCY	10
STEPWISE REFINEMENT	32
STESD	1
STRUCT	3
STRUCTURED DESIGN	40
STRUCTURED PROGRAMMING	181
STRUCTURED PROGRAMMING LANGUAGE	27
STRUCTURED WALK-THROUGH	9
STRUCTUREDNESS	10
STRUCTURE-CHANGING PROGRAMS	2

KEYWORD	NO. DOCUMENTS INDEXED USING THIS KEYWORD
SUBMARINE APPLICATIONS	5
SUPERVISORY PROGRAM	5
SYMBOLIC EXECUTION	21
SYNCHRONIZATION	27
SYNTAX GRAPHS	3
SYNTHESIZERS	1
SYSTEM ARCHITECTURE	24
SYSTEM DEPENDENCIES	1
SYSTEM DESIGN	104
SYSTEM DESIGN REQUIREMENTS	3
SYSTEM ENGINEERING LANGUAGE	6
SYSTEM INTEGRATION	4
SYSTEM MAINTENANCE	5
SYSTEM RELIABILITY	21
SYSTEM STRUCTURING	5
SYSTEM TESTING	28
SYSTEM VALIDATION	8
TACTICS	1
TARGET LANGUAGE	1
TECHNOLOGY TRANSFER	12
TELECOMMUNICATIONS APPLICATIONS	13
TERMINATION PROOFS	8
TEST DATA GENERATION	88
TEST DESIGN	10
TEST DRIVERS	11
TEST MANAGEMENT	26
TEST METHODOLOGIES	76
TEST PLAN DOCUMENT	27
TESTABILITY	11
TESTEDNESS	15
TESTING	115
TEXT-PROCESSING APPLICATIONS	15
TIME DOMAINS	16
TOOLS DIRECTORY	2

KEYWORD	NO. DOCUMENTS INDEXED USING THIS KEYWORD	KEYWORD	NO. DOCUMENTS INDEXED USING THIS KEYWORD
TOOLS TAXONOMY	1		
TOP DOWN DESIGN	39		
TOP DOWN DEVELOPMENT	19		
TOP-DOWN IMPLEMENTATION	5		
TOP-DOWN PARSING	4		
TOP-DOWN PROGRAMMING	24		
TOTAL CORRECTNESS	4		
TRAFFIC CONTROL APPLICATIONS	3		
TRANSFORMATION	1		
TRANSLATORS	2		
TRI-SERVICE	20		
TRW (SCEP) MODEL	1		
UNIX	22		
USABILITY	3		
USER-INTERACTIVE SYSTEMS	55		
UTILITY INDUSTRY APPLICATIONS	3		
VALIDATION	74		
VERIFICATION	99		
VERIFICATION TOOLS AND TECHNIQUES	70		
VIRTUAL MACHINE MONITOR	3		
VIRTUAL MACHINES	22		
WALK THROUGH	5		
WEAPONS SYSTEMS APPLICATIONS	13		
WELLMADE	1		
WORKING SETS	5		

APPENDIX D

STATISTICS ON KEYWORD DISTRIBUTION IN THE SERP DATABASE

APPENDIX D

STATISTICS ON KEYWORD DISTRIBUTION IN THE SERP DATABASE

This Appendix contains an alphabetical listing of 114 of the 519 keywords in the DACS Thesaurus together with the number of projects assigned each keyword. There were 524 keyword assignments made to 175 online project descriptions, yielding an average of three keywords per project.

KEYWORD	NO. PROJECTS INDEXED USING THIS KEYWORD
ALGORITHM ANALYSIS	1
APL (A PROGRAMMING LANGUAGE)	1
ARCHITECTURE	1
ARTIFICIAL INTELLIGENCE	2
AUDITS	1
AUTOMATED DESIGN TOOLS	2
AUTOMATED FAULT DETECTION	1
AUTOMATED TESTING	3
AUTOMATED VERIFICATION TOOLS	1
AUTOMATIC PROGRAMMING	3
COMPILER-COMPILERS	1
COMPUTER COMMUNICATIONS NETWORKS	1
CONFIGURATION MANAGEMENT	1
COST	2
COST AND SCHEDULE CONTROL	1
COST EFFECTIVENESS	2
COST ESTIMATION	4
COST/PRODUCTIVITY MODELS	2
DATA ANALYSIS	2
DATA COLLECTION	5
DATA REPOSITORY	1
DATABASE MANAGEMENT SYSTEMS	1
DEBUGGING	2
DESIGN	9
DESIGN ANALYSIS	1
DESIGN METHODOLOGIES	7
DESIGN TOOLS AND TECHNIQUES	6
DEVELOPMENT	5
DEVELOPMENT CYCLE	1
DEVELOPMENT MANAGEMENT	1
DEVELOPMENT SUPPORT LIBRARIAN	1
DEVELOPMENTAL METHODOLOGIES	3
DEVELOPMENTAL PROCESS	3
DEVELOPMENTAL TOOLS AND TECHNIQUES	1

KEYWORD	NO. PROJECTS INDEXED USING THIS KEYWORD
DISTRIBUTED PROCESSING	13
DOCUMENTATION	3
DOD COMMON HIGH ORDER LANGUAGE	5
EMBEDDED COMPUTER SYSTEMS	2
ERROR ANALYSIS	3
FAULT TOLERANCE	2
FAULT-TOLERANT SOFTWARE	1
FIRMWARE	1
FUNCTIONAL PROGRAMMING	1
HALSTEAD'S LAW	2
HIGHER-ORDER LANGUAGES	1
IMPLEMENTATION	1
LANGUAGE DESIGN	1
LANGUAGES	1
LIFE CYCLE COSTS	4
MAINTENANCE	3
MAINTENANCE TOOLS AND TECHNIQUES	3
MANAGEMENT	5
MANAGEMENT TOOLS AND TECHNIQUES	3
MATHEMATICAL METHODOLOGIES	2
MICRO COMPUTERS	1
MICROPROCESSORS	1
MODELLING AND SIMULATION TOOLS	1
MODELS	8
MODERN PROGRAMMING PRACTICES	4
MODIFICATION	1
MODULAR PROGRAMMING	2
OPERATIONAL RELIABILITY	1
PARTITIONING	2
PERFORMANCE	3
PERFORMANCE EVALUATION	2
PORTABILITY	1
PRODUCTIVITY FACTORS	1
PROGRAM ANALYSIS	1

KEYWORD	NO. PROJECTS INDEXED USING THIS KEYWORD
PROGRAM DESIGN LANGUAGE (PDL)	1
PROGRAM TESTING	3
PROGRAMMER PRODUCTIVITY	1
PROGRAMMING	2
PROGRAMMING AIDS	1
PROGRAMMING LANGUAGE	6
PROGRAMMING LANGUAGE COMPLEXITY	1
PROGRAMMING TECHNIQUES/METHODOLOGIES	2
PROTOCOLS	1
QUALITY ASSURANCE	6
QUALITY ATTRIBUTES	1
QUALITY METRICS	2
REAL-TIME SYSTEMS	1
RELIABILITY	25
RELIABILITY MODELS	1
REQUIREMENTS	21
REQUIREMENTS ANALYSIS	3
REQUIREMENTS ENGINEERING	4
REQUIREMENTS ENGINEERING METHODOLOGIES	1
REQUIREMENTS LANGUAGES	1
SOFTWARE DATABASE	1
SOFTWARE ENGINEERING	155
SOFTWARE ENGINEERING FACILITY	1
SOFTWARE ENGINEERING TOOLS AND TECHNIQUES	36
SOFTWARE LIFE CYCLE	2
SOFTWARE SCIENCE	3
SOFTWARE TOOLS	8
SPECIFICATIONS	5
STATIC ANALYSIS	1
STRUCTURED PROGRAMMING	4
SYSTEM ARCHITECTURE	4
SYSTEM DESIGN	4
SYSTEM RELIABILITY	2
SYSTEM TESTING	2
TEST DATA GENERATION	2
TEST METHODOLOGIES	4
TESTEDNESS	1
TESTING	8
TOP DOWN DESIGN	1
TOP DOWN DEVELOPMENT	1
UNIX	1
USER-INTERACTIVE SYSTEMS	1
VALIDATION	7
VERIFICATION	17
VERIFICATION TOOLS AND TECHNIQUES	2
VFTT	1

MISSION
of
Rome Air Development Center

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control Communications and Intelligence (C^3I) activities. Technical and engineering support within areas of technical competence is provided to ESD Program Offices (POs) and other ESD elements. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.


DAT
ILM